Package ‘pgam’

January 24, 2018

Version 0.4.15
Date 2018-01-25
Author Washington Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>
Maintainer Washington Junger <wjunger@ims.uerj.br>
Depends R (>= 3.0.0),stats,utils
Title Poisson-Gamma Additive Models
Description This work is an extension of the state space model for Poisson count data, Poisson-Gamma model, towards a semiparametric specification. Just like the generalized additive models (GAM), cubic splines are used for covariate smoothing. The semiparametric models are fitted by an iterative process that combines maximization of likelihood and backfitting algorithm.
License GPL (>= 2)
NeedsCompilation yes
Repository CRAN
Date/Publication 2018-01-24 20:55:01 UTC

R topics documented:

AIC.pgam .......................................................... 2
aihrio ............................................................. 3
coef.pgam ......................................................... 5
deviance.pgam .................................................. 6
envelope.pgam ................................................... 7
f ................................................................. 8
fitted.pgam ....................................................... 9
g ............................................................... 10
logLik.pgam ..................................................... 11
periodogram ...................................................... 12
pgam ............................................................ 13
plot.pgam ......................................................... 15
predict.pgam .................................................... 16
print.pgam ....................................................... 18
Description

Method for approximate Akaike Information Criterion extraction.

Usage

```r
## S3 method for class 'pgam'
AIC(object, k = 2, ...)
```

Arguments

- `object`: object of class `pgam` holding the fitted model
- `k`: default is 2 for AIC. If \( k = \log(n) \) then an approximation for BIC is obtained. Important to note that these are merely approximations.
- `...`: further arguments passed to method

Details

An approximate measure of parsimony of the Poisson-Gama Additive Models can be achieved by the expression

\[
AIC = \frac{D(y; \hat{\mu}) + 2gle}{n - \tau}
\]

where \( gle \) is the number of degrees of freedom of the fitted model and \( \tau \) is the index of the first non-zero observation.

Value

The approximate AIC value of the fitted model.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

References


aihrio

See Also
gam, deviance.gam, logLik.gam

Examples

library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP5~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form, aihrio, omega=.8, beta=.01, maxit=1e2, eps=1e-4, optim.method="BFGS")

AIC(m)

---

aihrio  Sample dataset

Description

This is a dataset for Poisson-Gamma Additive Models functions testing.

Usage

data(aihrio)

Format

A data frame with 365 observations on the following 33 variables.

- DATE  a factor with levels
- TIME  a numeric vector
- ITRESP65  a numeric vector
- ITCIRC65  a numeric vector
- ITDPOC65  a numeric vector
- ITPNM65  a numeric vector
- ITAVC65  a numeric vector
- ITIAM65  a numeric vector
- ITDIC65  a numeric vector
- ITTCA65  a numeric vector
- ITRESP5  a numeric vector
- ITPNEU5  a numeric vector
- ITDPC5  a numeric vector
- WEEK  a numeric vector
- MON  a numeric vector
TUE  a numeric vector
WED  a numeric vector
THU  a numeric vector
FRI  a numeric vector
SAT  a numeric vector
SUN  a numeric vector
HOLIDAYS a numeric vector
MONTH a numeric vector
warm.season a numeric vector
tmpmed a numeric vector
tmpmin a numeric vector
tmpmax a numeric vector
wet  a numeric vector
rain  a numeric vector
rainy a numeric vector
PM   a numeric vector
SO2  a numeric vector
CO   a numeric vector

Details

This is a reduced dataset of those used to estimate possible effects of air pollution on hospital admissions outcomes in Universidade do Estado do Rio de Janeiro, Rio de Janeiro, Brasil.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

Source

Secretary for the Environment of the Rio de Janeiro City, Brazilian Ministry of Defense and Brazilian Ministry of Health
coef.pgam

Coefficients extraction

Description

Method for parametric coefficients extraction.

Usage

## S3 method for class 'pgam'
coef(object, ...)

Arguments

- **object**: object of class pgam holding the fitted model
- **...**: further arguments passed to method

Details

This function only retrieves the estimated coefficients from the model object returned by pgam.

Value

Vector of coefficients estimates of the model fitted.

Author(s)

Washington Leite Junger <wjuner@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

References


See Also

pgam, pgam.fit, predict.pgam

Examples

library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP5~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form, aihrio, omega=.8, beta=.01, maxit=1e2, eps=1e-4, optim.method="BFGS")
### deviance.pgam

**Description**

Method for total deviance value extraction.

**Usage**

```r
## S3 method for class 'pgam'
deviance(object, ...)
```

**Arguments**

- `object`: object of class `pgam` holding the fitted model
- `...`: further arguments passed to method

**Details**

See `predict.pgam` for further information on deviance extraction in Poisson-Gamma models.

**Value**

The sum of deviance components.

**Author(s)**

Washington Leite Junger `<wjunger@ims.uerj.br>` and Antonio Ponce de Leon `<ponce@ims.uerj.br>`

**References**


**See Also**

`pgam, pgam.fit, pgam.likelihood`
Examples

```r
library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP5~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
mod <- pgam(form,aihrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")

deviance(mod)
```

envelope.pgam

A normal plot with simulated envelope of the residuals.

Description

A normal plot with simulated envelope of the residual is produced.

Usage

```r
## S3 method for class 'pgam'
envelope(object, type = "deviance", size = 0.95,
rep = 19, optim.method = NULL, epsilon = 0.001, maxit = 100,
plot = TRUE, title="Simulated Envelope of Residuals", verbose = FALSE, ...)
```

Arguments

- `object`: object of class `pgam` holding the fitted model
- `type`: type of residuals to be extracted. Default is `deviance`. Options are described in `residuals.pgam`
- `size`: value giving the size of the envelope. Default is .95 which is equivalent to a 95% band
- `rep`: number of replications for envelope construction. Default is 19, that is the smallest 95% band that can be build
- `optim.method`: optimization method to be passed to `pgam` and therefore to `optim`
- `epsilon`: convergence control to be passed to `pgam`
- `maxit`: convergence control to be passed to `pgam`
- `plot`: if `TRUE` a plot of the envelope is produced
- `title`: title for the plot
- `verbose`: if `TRUE` a sort of information is printed during the running time
- `...`: further arguments to `plot` function
Details

Method for the generic function `envelope`.

Sometimes the usual Q-Q plot shows an unsatisfactory pattern of the residuals of a model fitted and we are led to think that the model is badly specified. The normal plot with simulated envelope indicates that under the distribution of the response variable the model is OK if only a few points fall off the envelope.

If object is of class `pgam` the envelope is estimated and optionally plotted, else if is of class `envelope` then it is only plotted.

Value

An object of class `envelope` holding the information needed to plot the envelope.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

References


See Also

`pgam`, `predict.pgam`, `residuals.pgam`

---

**f**

<table>
<thead>
<tr>
<th>Utility function</th>
</tr>
</thead>
</table>

**Description**

Generate the partition of design matrix regarded to the seasonal factor in its argument. Used in the model formula.

**Usage**

`f(factorvar)`

**Arguments**

- `factorvar` variable with the seasonal levels

**Value**

List containing data matrix of dummy variables, level names and seasonal periods.
fitted.pgam

Note

This function is intended to be called from within a model formula.

Author(s)

Washington Leite Junger <wjuner@ims.uerj.br>

See Also

pgam, formparser

fitted.pgam  Fitted values extraction

Description

Method for fitted values extraction.

Usage

## S3 method for class 'pgam'
fitted(object, ...)

Arguments

object  object of class pgam holding the fitted model

...  further arguments passed to method

Details

Actually, the fitted values are worked out by the function predict.pgam. Thus, this method is supposed to turn fitted values extraction easier. See predict.pgam for details on one-step ahead prediction.

Value

Vector of predicted values of the model fitted.

Author(s)

Washington Leite Junger <wjuner@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

References


See Also

`pgam`, `pgamNfit`, `predictNpgam`  

Examples

```r
library(pgam)
data(aishrio)
attach(aishrio)
form <- ITRESP~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form,aishrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")
f <- fitted(m)
```

---

**g**  

*Utility function*

Description

Collect information to smooth the term in its argument. Used in the model formula.

Usage

```r
g(var, df = NULL)
```

Arguments

- `var`: variable to be smoothed
- `df`: equivalent degrees of freedom to be passed to the smoother. If `NULL`, smoothing parameter is selected by cross-validation

Details

This function only sets things up for model fitting. The smooth terms are actually fitted by `bkfsmooth`.

Value

List containing the same elements of its argument.

Note

This function is intended to be called from within a model formula.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br>
References


See Also

\texttt{pgam, formparser}

---

\textbf{Description}

Method for loglik value extraction.

\textbf{Usage}

```r
## S3 method for class ‘pgam’
logLik(object, ...)
```

\textbf{Arguments}

- \texttt{object}:
  object of class \texttt{pgam} holding the fitted model

- \texttt{...}:
  further arguments passed to method

\textbf{Details}

See \texttt{pgam.likelihood} for more information on log-likelihood evaluation in Poisson-Gamma models.

\textbf{Value}

The maximum value achieved by the likelihood optimization process.

\textbf{Author(s)}

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

\textbf{References}


See Also

pgam, pgam.fit.pgam.likelihood

Examples

library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form, aihrrio, omega=.8, beta=.01, maxit=1e2, eps=1e-4, optim.method="BFGS")

logLik(m)

<table>
<thead>
<tr>
<th>periodogram</th>
<th>Raw Periodogram</th>
</tr>
</thead>
</table>

Description

A raw periodogram is returned and optionally plotted.

Usage

periodogram(y, rows = trunc(length(na.omit(y))/2-1), plot = TRUE, ...)

Arguments

- **y**
  - time series
- **rows**
  - number of rows to be returned. Default and largest is \( n/2 - 1 \), where \( n \) is the number of valid observations of the time series \( y \)
- **plot**
  - if TRUE a raw periodogram is plotted
- **...**
  - further arguments to `plot` function

Details

The raw periodogram is an estimator of the spectrum of a time series, it still is a good indicator of unresolved seasonality patterns in residuals of the fitted model. Check the function `intensity` for frequencies extraction.

This function plots a fancy periodogram where the intensities of the angular frequencies are plotted resembling tiny lollipops.

Value

Periodogram ordered by intensity.
Description

Fit Poisson-Gamma Additive Models using the roughness penalty approach

Usage

```r
pgam(formula, dataset, omega = 0.8, beta = 0.1, offset = 1, digits =getOption("digits"),
      na.action="na.exclude", maxit = 100, eps = 1e-06, lfn.scale=1, control = list(),
      optim.method = "L-BFGS-B", bkf.eps = 0.001, bkf.maxit = 100, se.estimation = "numerical",
      verbose = TRUE)
```

Arguments

- **formula**: a model formula. See `formparser` for details
- **dataset**: a data set in the environment search path. Missing data is temporarily not handled
- **omega**: initial value for the discount factor
- **beta**: vector of initial values for covariates coefficients. If a single value is supplied it is replicated to fill in the whole vector
- **offset**: default is 1. Other value can be supplied here
- **digits**: number of decimal places for printing information out
- **na.action**: action to be taken if missing values are found. Default is "na.exclude" and residuals and predictions are padded to fit the length of the data. If "na.fail" then the process will stop if missing values are found. If "na.omit" the process will continue without padding though. If "na.pass" the process will stop due to errors
- **maxit**: convergence control iterations
- **eps**: convergence control criterion
lfn.scale scales the likelihood function and is passed to control in optim. Value must be positive to ensure maximization

control convergence control of optim. See its help for details

optim.method optimization method passed to optim. Different methods can lead to different results, so the user must attempt to the trade off between speed and robustness. For example, BFGS is faster but sensitive to starting values and L-BFGS-B is more robust but slower. See its help for details.

bkf.eps convergence control criterion for the backfitting algorithm

bkf.maxit convergence control iterations for the backfitting algorithm

se.estimation if numerical standard error of parameters are returned. If analytical then analytical extraction of the standard errors is performed. By setting it to none standard error estimation is avoided

verbose if TRUE information during estimation process is printed out

Details

The formula is parsed by formparser in order to extract all the information necessary for model fit. Split the model into two parts regarding the parametric nature of the model. A model can be specified as following:

\[ Y \ f(s_{f_r}) + V1 + V2 + V3 + g(V4, df4) + g(V5, df5) \]

where \( s_{f_r} \) is a seasonal factor with period \( r \) and \( df_i \) is the degree of freedom of the smoother of the \( i \)-th covariate. Actually, two new formulae will be created:

\[ s_{f_1} + \ldots + s_{f_r} + V1 + V2 + V3 \]

and

\[ V4 + V5 \]

These two formulae will be used to build the necessary datasets for model estimation. Dummy variables reproducing the seasonal factors will be created also.

Models without explanatory variables must be specified as in the following formula

\[ Y \ NULL \]

There are a lot of details to be written. It will be very soon. Specific information can be obtained on functions help.

This algorithm fits fully parametric Poisson-Gamma model also.

Value

List containing an object of class pgam.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>
References


See Also

predict.pgam, formparser, residuals.pgam, backfitting

Examples

library(pgam)
data(aihrio)
attach(aihrio)
form <- ~ ITRESP5*f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form,aihrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")

summary(m)

plot.pgam  Plot of estimated curves

Description

Plot of the local level and, when semiparametric model is fitted, the estimated smooth terms.

Usage

## S3 method for class 'pgam'
plot(x, rug = TRUE, se = TRUE, at.once = FALSE, scaled = FALSE, ...)

Arguments

x object of class pgam holding the fitted model
rug if TRUE a density rug is drawn on the bottom of the graphic
se if TRUE error band is drawn around the fitted values
at.once if TRUE each plot goes to a separate window, else the user is prompted to continue
scaled if TRUE the same scale will be used for plots of smoothed functions
... further arguments passed to method
**predict.pgam**

**Details**

Error band of smooth terms is approximated.

**Value**

No value returned.

**Author(s)**

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

**See Also**

pgam, pgam.fit, pgam.likelihood

**Examples**

```r
library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form,aihrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")
plot(m,at.once=TRUE)
```

---

**Description**

Prediction and forecasting of the fitted model.

**Usage**

```r
## S3 method for class 'pgam'
predict(object, forecast = FALSE, k = 1, x = NULL, ...)
```

**Arguments**

- `object`: object of class pgam holding the fitted model
- `forecast`: if TRUE the function tries to forecast
- `k`: steps for forecasting
- `x`: covariate values for forecasting if the model has covariates. Must have the $k$ rows and $p$ columns
- `...`: further arguments passed to method
**Details**

It estimates predicted values, their variances, deviance components, generalized Pearson statistics components, local level, smoothed prediction and forecast.

Considering a Poisson process and a gamma prior, the predictive distribution of the model is negative binomial with parameters $a_{t|t-1}$ and $b_{t|t-1}$. So, the conditional mean and variance are given by

$$E(y_t|Y_{t-1}) = a_{t|t-1}/b_{t|t-1}$$

and

$$Var(y_t|Y_{t-1}) = a_{t|t-1}(1+b_{t|t-1})/b_{t|t-1}^2$$

Deviance components are estimated as follow

$$D(y; \hat{\mu}) = 2 \sum_{t=\tau+1}^{n} a_{t|t-1} \log \left( \frac{a_{t|t-1}}{y_t b_{t|t-1}} \right) - (a_{t|t-1} + y_t) \log \left( \frac{y_t + a_{t|t-1}}{1 + b_{t|t-1}} \right) y_t$$

Generalized Pearson statistics has the form

$$X^2 = \sum_{t=\tau+1}^{n} \frac{(y_t b_{t|t-1} - a_{t|t-1})^2}{a_{t|t-1}(1+b_{t|t-1})}$$

Approximate scale parameter is given by the expression

$$\hat{\phi} = \frac{X^2}{edf}$$

where $edf$ is the number of degrees of freedom of the fitted model.

**Value**

List with those described in Details

**Author(s)**

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

**References**


See Also

\texttt{pgam.residuals.pgam}

Examples

\begin{verbatim}
library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP5~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form,aihrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")

p <- predict(m)$yhat
plot(ITRESP5)
lines(p)
\end{verbatim}

Description

Print model information

Usage

\begin{verbatim}
## S3 method for class 'pgam'
print(x, digits, ...)
\end{verbatim}

Arguments

- \texttt{x} \hspace{1cm} object of class \texttt{summary.pgam} holding the fitted model information
- \texttt{digits} \hspace{1cm} number of decimal places for output
- \texttt{...} \hspace{1cm} further arguments passed to method

Details

This function only prints out the information.

Value

No value is returned.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

See Also

\texttt{pgam, predict.pgam}
**print.summary.pgam**

Summary output

**Description**
Print output of model information

**Usage**
```r
## S3 method for class 'pgam'
print.summary(x, digits, ...)
```

**Arguments**
- `x`: object of class `summary.pgam` holding the fitted model information
- `digits`: number of decimal places for output
- `...`: further arguments passed to method

**Details**
This function actually only prints out the information.

**Value**
No value is returned.

**Author(s)**
Washington Leite Junger \(<wjunger@ims.uerj.br>\) and Antonio Ponce de Leon \(<ponce@ims.uerj.br>\)

**See Also**
- `pgam`
- `predict.pgam`

**residuals.pgam**

Residuals extraction

**Description**
Method for residuals extraction.

**Usage**
```r
## S3 method for class 'pgam'
residuals(object, type = "deviance", ...)
```
Arguments

- **object**: object of class `pgam` holding the fitted model
- **type**: type of residuals to be extracted. Default is `deviance`. Options are described in Details.

... further arguments passed to method

Details

The types of residuals available and a brief description are the following:

- **response**: These are raw residuals of the form $r_t = y_t - E(y_t|Y_t-1)$.
- **pearson**: Pearson residuals are quite known and for this model they take the form $r_t = \frac{y_t - E(y_t|Y_t-1)}{\text{Var}(y_t|Y_t-1)}$.
- **deviance**: Deviance residuals are estimated by $r_t = \text{sign}(y_t - E(y_t|Y_t-1)) \times \sqrt{d_t}$, where $d_t$ is the deviance contribution of the $t$-th observation. See `deviance.pgam` for details on deviance component estimation.
- **std_deviance**: Same as deviance, but the deviance component is divided by $(1 - h_t)$, where $h_t$ is the $t$-th element of the diagonal of the pseudo hat matrix of the approximating linear model. So they turn into $r_t = \text{sign}(y_t - E(y_t|Y_t-1)) \times \sqrt{d_t / (1 - h_t)}$.

The element $h_t$ has the form $h_t = \omega \exp(\eta_{t+1}) / \sum_{j=0}^{t-1} \omega \exp(\eta_{t-j})$, where $\eta$ is the predictor of the approximating linear model.

- **std_scl_deviance**: Just like the last one except for the dispersion parameter in its expression, so they have the form $r_t = \text{sign}(y_t - E(y_t|Y_t-1)) \times \sqrt{d_t / \phi \times (1 - h_t)}$, where $\phi$ is the estimated dispersion parameter of the model. See `summary.pgam` for $\phi$ estimation.

Value

Vector of residuals of the model fitted.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>

References


See Also

`pgam`, `pgam.fit`, `predict.pgam`

Examples

```r
library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP~f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form,aihrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")

r <- resid(m,"pearson")
plot(r)
```

**summary.pgam**

Summary output

### Description

Output of model information

### Usage

```r
## S3 method for class 'pgam'
summary(object, smo.test = FALSE, ...)
```

### Arguments

- `object` object of class pgam holding the fitted model
- `smo.test` Approximate significance test of smoothing terms. It can take long, so default is FALSE
- `...` further arguments passed to method

### Details

Hypothesis tests of coefficients are based on $t$ distribution. Significance tests of smooth terms are approximate for model selection purpose only. Be very careful about the later.

### Value

List containing all the information about the model fitted.

### Author(s)

Washington Leite Junger <wjunger@ims.uerj.br> and Antonio Ponce de Leon <ponce@ims.uerj.br>
References


See Also

pgam, predict.pgam

Examples

library(pgam)
data(aihrio)
attach(aihrio)
form <- ITRESP5+f(WEEK)+HOLIDAYS+rain+PM+g(tmpmax,7)+g(wet,3)
m <- pgam(form,aihrio,omega=.8,beta=.01,maxit=1e2,eps=1e-4,optim.method="BFGS")

summary(m)

tbl2tex

LaTeX table exporter

Description

Export a data frame to a fancy LaTeX table environment.

Usage

tbl2tex(tbl, label = "tbl:label(must_be_changed!)", caption = "Table generated with tbl2tex.", centered = TRUE, alignment = "center", digits = getOption("digits"), hline = TRUE, vline = TRUE, file = "", topleftcell = "")
Arguments

- tbl: object of type data frame or matrix
- label: label for LaTeX cross reference
- caption: caption for LaTeX tabular environment
- centered: logical. TRUE for centered cells
- alignment: alignment of the object on the page
- digits: decimal digits after decimal point
- hline: logical. TRUE for horizontal borders
- vline: logical. TRUE for vertical borders
- file: filename for outputting. If none is provided, LaTeX code is routed through the console
- topleftcell: text for the top-left cell of the table

Details

This is a utility function intended to ease conversion of R objects to LaTeX format. It only exports data frame or data matrix nonetheless.

Value

LaTeX code is routed through file or console for copying and pasting.

Note

For now, it handles only numerical data.

Author(s)

Washington Leite Junger <wjunger@ims.uerj.br>

See Also

pgam

Examples

library(pgam)
data(aihrio)
m <- aihrio[1:10,4:10]
tbl2tex(m,label="tbl:r_example",caption="R example of tbl2tex",digits=4)
Index

*Topic **datasets**
  aihrio, 3

*Topic **regression**
  AIC.pgam, 2
  coef.pgam, 5
  deviance.pgam, 6
  envelope.pgam, 7
  f, 8
  fitted.pgam, 9
  g, 10
  logLik.pgam, 11
  periodogram, 12
  pgam, 13
  plot.pgam, 15
  predict.pgam, 16
  print.pgam, 18
  print.summary.pgam, 19
  residuals.pgam, 19
  summary.pgam, 21
  tbl2tex, 22

*Topic **smooth**
  AIC.pgam, 2
  coef.pgam, 5
  deviance.pgam, 6
  envelope.pgam, 7
  f, 8
  fitted.pgam, 9
  g, 10
  logLik.pgam, 11
  periodogram, 12
  pgam, 13
  plot.pgam, 15
  predict.pgam, 16
  print.pgam, 18
  print.summary.pgam, 19
  residuals.pgam, 19
  summary.pgam, 21
  tbl2tex, 22

*Topic **ts**

AIC.pgam, 2
coef.pgam, 5
deviance.pgam, 6
envelope.pgam, 7
f, 8
fitted.pgam, 9
g, 10
logLik.pgam, 11
periodogram, 12
pgam, 13
plot.pgam, 15
predict.pgam, 16
print.pgam, 18
print.summary.pgam, 19
residuals.pgam, 19
summary.pgam, 21
tbl2tex, 22

backfitting, 15
bkfsMOOTH, 10

coefficient, 5
deviance.pgam, 3, 6, 20
envelope, 8
envelope.pgam, 7
f, 8
fitted.pgam, 9
formparser, 9, 11, 13–15

logLik.pgam, 3, 11
optim, 7, 14

periodogram, 12
INDEX

pgam, 3, 5–13, 16, 18, 19, 21–23
pgam.fit, 5, 6, 10, 12, 16, 21
pgam.likelihood, 6, 11, 12, 16
plot, 7, 12
plot.pgam, 15
predict.pgam, 5, 6, 8–10, 15, 16, 18, 19, 21, 22
print.pgam, 18
print.summary.pgam, 19
residuals.pgam, 7, 8, 15, 18, 19
summary.pgam, 20, 21
tbl2tex, 22