# Package ‘COZIGAM’

January 2, 2012

**Type** Package

**Title** Constrained and Unconstrained Zero-Inflated Generalized Additive Models with Model Selection Criterion

**Version** 2.0-3

**Date** 2010-7-23

**Author** Hai Liu <liuhai@iupui.edu>, Kung-Sik Chan <kung-sik-chan@uiowa.edu>

**Maintainer** Hai Liu <liuhai@iupui.edu>

**Depends** R (>= 2.7.1), mgcv

**Description** Constrained and Unconstrained Zero-Inflated Generalized Additive Models (ZIGAM) fitting with associated model plotting,prediction and selection.

**License** GPL (>= 2)

**Repository** CRAN

**Date/Publication** 2010-07-24 07:57:05

## R topics documented:

---

`COZIGAM-package` *Constrained and Unconstrained Zero-Inflated Generalized Additive Models with Model Selection Criterion*

---

## Description

Constrained and Unconstrained Zero-Inflated Generalized Additive Models (ZIGAM) fitting with associated model plotting, prediction and selection.

## Details

| | |
|---|---|
| Package: | COZIGAM |
| Type: | Package |
| Version: | 2.0-3 |
| Date: | 2010-7-23 |
| License: | GPL version 2 or newer |

| | |
|---|---|
| COZIGAM.cts | Fitting Component-Specific-Proportionally Constrained Zero-Inflated Generalized Additive Models with |
| COZIGAM.dis | Fitting Component-Specific-Proportionally Constrained Zero-Inflated Generalized Additive Models with |
| PCOZIGAM.cts | Fitting Proportionally Constrained Zero-Inflated Generalized Additive Models with Continuous Respons |
| PCOZIGAM.dis | Fitting Proportionally Constrained Zero-Inflated Generalized Additive Models with Discrete Responses |
| ZIGAM.cts | Fitting Continuous Zero-Inflated Generalized Additive Models |
| ZIGAM.dis | Fitting Discrete Zero-Inflated Generalized Additive Models |
| alaska | Alaska Peninsula Coastline |
| cozigam | Fitting Constrained Zero-Inflated Generalized Additive Models |
| disgam | Fitting Discrete Generalized Additive Models with Model Selection Criterion |
| eggdata | Pollock Egg Data |
| plot.cozigam | Default COZIGAM plotting |
| predict.cozigam | Prediction from fitted COZIGAM |
| print.cozigam | Constrained Zero-Inflated Generalized Additive Model default print statement |
| summary.cozigam | Summary for a COZIGAM fit |
| test | Some Test Functions used in Simulation Study |
| zigam | Fitting (Unconstrained) Zero-Inflated Generalized Additive Models |

### Author(s)

Hai Liu and Kung-Sik Chan; Maintainer: Hai Liu <liuhai@iupui.edu>

---

`alaska` *Alaska Peninsula Coastline*

---

### Description

The coastline of the Alaska Peninsula.

### Usage

```
data(alaska)
```

### Format

A data frame with the following 2 variables.

`lat` A numeric vector: latitude

`lon` A numeric vector: longitude

### Details

This data frame can be used to draw the coastline of the Alaska peninsula. See example below.

### Author(s)

Hai Liu and Kung-Sik Chan

### Examples

```
data(alaska)
plot(alaska$lon,alaska$lat,type="l",col="green")
```

cozigam *Fitting Constrained Zero-Inflated Generalized Additive Models*

## Description

Fit a COnstrained Zero-Inflated Generalized Additive Model (COZIGAM) to data.

## Usage

```
cozigam (formula, constraint = "proportional", zero.delta = NULL,
    maxiter = 20, conv.crit.in = 1e-5, conv.crit.out = 1e-3, size = NULL,
    log.tran = FALSE, family, data=list(), ...)
```

## Arguments

| | |
|---|---|
| `formula` | A GAM formula. This is exactly like the formula for a GLM except that smooth terms can be added to the right hand side of the formula (and a formula of the form `y ~ .` is not allowed). Smooth terms are specified by expressions of the form: `s(var1,var2,...,k=12,fx=FALSE,bs="tp",by=a.var)` where `var1`, `var2`, etc. are the covariates which the smooth is a function of and `k` is the dimension of the basis used to represent the smooth term. `by` can be used to specify a variable by which the smooth should be multiplied. |
| `constraint` | Type of constraint on the zero-inflation probability, can be either “proportional" or “component". See details for more information. |
| `zero.delta` | A vector specifying which subset of constraint parameter delta set to be zero when the `constraint` is set to “component". For instance, `zero.delta=c(NA, 0)` means only the first smooth function is included in the zero-inflation constraint. See details for the model formulation. |
| `maxiter` | The maximum number of iterations allowed in the estimation procedure. |
| `conv.crit.in` | Convergence criterion in the inner loop. |
| `conv.crit.out` | Convergence criterion in the outer loop. |
| `size` | Number of trials. Must be specified when `family` is `binomial`. |
| `log.tran` | Logical. `TRUE` if log-transformation is needed for the response. |
| `family` | This is a family object specifying the distribution and link to use in fitting etc. See `glm` and `family` for more details. Currently support Gaussian/lognormal, Gamma, Poisson and binomial distributions. |
| `data` | A data frame or list containing the model response variable and covariates required by the formula. |
| `...` | Additional arguments to be passed to the low level regression fitting functions. |

## Details

A COnstrained Zero-Inflated Generalized Additive Model (COZIGAM) assumes the response variable $Y_i$ is distributed from a zero-inflated 1-parameter exponential family with covariate $x_i$ (could be high dimensional). More specifically, $Y_i$ comes from a regular exponential family distribution $f(x_i)$ with probability $p_i$ and equals zero with probability $1 - p_i$, with the further assumption that the probability of non-zero-inflation $p_i$ is related to the regular mean response or its smooth components, depending on the type of the `constraint`. If `constraint` is “proportional", $p_i$ is assumed to be some monotone function of the regular mean response function on the link scale, e.g., if a logit link is used, $p_i$ is linearly constrained by:

$$logit(p_i) = \alpha + \delta g(\mu_i),$$

where $g()$ is the link function of the regular exponential family with mean $\mu_i$ and $\alpha$ and $\delta$ are two unknown parameters to be estimated; If `constraint` is “component", $p_i$ is assumed to be linearly related to the smooth components of the regular mean response on the link scale, i.e.,

$$logit(p_i) = \alpha + \delta_1 s(x_{1i}, x_{2i}) + \delta_2 s(x_{3i}) + ...,$$

and the regular mean response has the structure

$$g(\mu_i) = b_0 + s(x_{1i}, x_{2i}) + s(x_{3i}) + ...,$$

where $s()$'s are some centered smooth functions to be estimated nonparametrically by a Generalized Additive Model (GAM) and $b_0$ is the overall intercept. In the component-specific-proportional constraint case, a vector `zero.delta` must be specified. For instance, in the above model setting, `zero.delta=c(NA, 0)` would estimate $\delta_1$ as an unknown parameter and set $\delta_2$ to be fixed at 0. That is, the model would assume that $p_i$ is not dependent on the covariate $x_3$.

The estimation approach is a modified penalized iteratively reweighted least squares algorithm (P-IRLS) which requires the `magic()` function in the `mgcv` library. EM algorithm is used if the underlying regular exponential family has strictly positive probability mass at zero. The covariance matrix of the estimated coefficients is obtained by inverting the observed information matrix. See Liu and Chan (2010) for more detail.

## Value

An object of class `"cozigam"` is a list containing the following components:

`coefficients` A named vector of coefficients including the linear constraint parameters.

`V.theta` The covariance matrix of the estimated parameters.

`V.beta` The covariance matrix of the estimated parameters associated with the smooth functions.

`se.alpha, se.delta` Estimated standard errors of parameters alpha and delta.

`mu` The fitted regular mean values.

`linear.predictor` The fitted values on the link scale.

`dispersion` (Estimated) dispersion parameter.

`formula` Model formula.

p The fitted non-zero-inflation probabilities.

G An objected returned by gam() containing information used in the estimation procedure.

psi Conditional expectation of the zero-inflation indicator (only for the discrete case which involves EM algorithm).

family The family used.

loglik, ploglik
The (penalized) log-likelihood of the fitted model.

y The response used.

converged Logical. TRUE if the iterative procedure is converged.

est.disp Logical. TRUE if the dispersion parameter is estimated.

fit.nonzero Fitted GAM using only the nonzero data as in the presence/absence analysis

score The GCV or UBRE score returned by magic() in the iterative procedure.

edf Estimated degrees of freedom for each model parameter. Penalization means that many of these are less than 1.

edf.smooth Estimated degrees of freedom of each smooth term.

sp Estimated smoothing parameters.

logE Approximated logarithmic marginal likelihood by Laplace method used for model selection.

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

plot.cozigam, predict.cozigam, summary.cozigam, zigam

## Examples

```
## Normal/Log-Normal Response with proportional constraint
set.seed(11)
n <- 400
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)

f <- test(x1,x2)*4-mean(test(x1,x2)*4) + f0(x3)/2-mean(f0(x3)/2)
sig <- 0.5
mu0 <- f + 3
```

```
y <- mu0 + rnorm(n, 0, sig)

alpha0 <- -2.2
delta0 <- 1.2
p0 <- .Call("logit_linkinv", alpha0 + delta0 * mu0, PACKAGE = "stats")
z <- rbinom(rep(1,n), 1, p0)
y[z==0] <- 0

res <- cozigam(y~s(x1,x2)+s(x3), constraint = "proportional", family = gaussian)

plot(res)

## Poisson Response with component-specific constraint
set.seed(11)
n <- 600
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)

f <- test(x1, x2)*2 + f0(x3)/5
eta0 <- f/1.1
mu0 <- exp(eta0)

eta.p10 <- (test(x1,x2) - mean(test(x1,x2)))*2/1.1
eta.p20 <- (f0(x3) - mean(f0(x3)))/5/1.1

alpha0 <- 0.2
delta10 <- 1.2
delta20 <- 0
eta.p0 <- delta10*eta.p10 + delta20*eta.p20
p0 <- .Call("logit_linkinv", alpha0 + eta.p0, PACKAGE = "stats")

z <- rbinom(rep(1,n), 1, p0)
y <- rpois(rep(1,n), mu0)
y[z==0] <- 0; rm(z)

res <- cozigam(y~s(x1,x2)+s(x3), constraint="component", zero.delta=c(NA, 0),
  conv.crit.out = 1e-3, family=poisson)


## A Tensor Product Smooth Example
test.ten <- function(x,z,sx=0.3,sz=0.4)
{ x<-x*20
  (pi**sx*sz)*(1.2*exp(-(x-0.2)^2/sx^2-(z-0.3)^2/sz^2)+
  0.8*exp(-(x-0.7)^2/sx^2-(z-0.8)^2/sz^2))
}

set.seed(11)
n <- 400
x1 <- runif(n, 0, 1)/20
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)
```

```
f <- test.ten(x1, x2)*4-mean(test.ten(x1, x2)*4) + f0(x3)/2-mean(f0(x3)/2)
sig <- 0.5
mu0 <- f + 3
y <- mu0 + rnorm(n, 0, sig)

alpha0 <- -2.2
delta0 <- 1.2
p0 <- .Call("logit_linkinv", alpha0 + delta0 * mu0, PACKAGE = "stats")

z <- rbinom(rep(1,n), 1, p0)
y[z==0] <- 0

# If use thin plate spline ...
res.tps <- cozigam(y~s(x1,x2)+s(x3), constraint = "proportional", family=gaussian)
par(mfrow=c(1,2))
plot(res.tps, select=1)
# Compare with tensor product spline
res.ten <- cozigam(y~te(x1,x2)+s(x3), constraint = "proportional", family=gaussian)
plot(res.ten, select=1)
```

---

## COZIGAM.cts — *Fitting Component-Specific-Proportionally Constrained Zero-Inflated Generalized Additive Models with Continuous Responses*

---

### Description

`COZIGAM.cts` fits a COZIGAM with component-specific-proportional constraint for continuous exponential family responses.

`COZIGAM.dis` fits a COZIGAM with discrete exponential family responses. One of the two funcions is called from `cozigam` automatically depending on the distribution family if the argument `constraint="component"`.

### Author(s)

Hai Liu and Kung-Sik Chan

### References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

### See Also

`cozigam`

COZIGAM.dis *Fitting Component-Specific-Proportionally Constrained Zero-Inflated Generalized Additive Models with Discrete Responses*

## Description

`COZIGAM.dis` fits a COZIGAM with component-specific-proportional constraint for discrete zero-inflated exponential family responses, e.g. Poisson and binomial distributions. If the regular exponential family admits 0 as a possible realization with positive probability, the EM algorithm is used in the estimation procedure.

`COZIGAM.cts` fits a COZIGAM with continuous zero-inflated exponential family responses. One of the two funcions is called from `cozigam` automatically depending on the distribution family if the argument `constraint="component"`.

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

`cozigam`

disgam *Fitting Discrete Generalized Additive Models with Model Selection Criterion*

## Description

Fit a discrete Generalized Additive Model (GAM) to data and calculate the logarithmic marginal likelihood.

## Usage

```
disgam (formula, size=NULL, family = poisson(), data=list(), ...)
```

## Arguments

| | |
|---|---|
| `formula` | A GAM formula. This is exactly like the formula for a GLM except that smooth terms can be added to the right hand side of the formula (and a formula of the form `y ~ .` is not allowed). Smooth terms are specified by expressions of the form: `s(var1,var2,...,k=12,fx=FALSE,bs="tp",by=a.var)` where `var1`, `var2`, etc. are the covariates which the smooth is a function of and `k` is the dimension of the basis used to represent the smooth term. `by` can be used to specify a variable by which the smooth should be multiplied. |
| `size` | Number of trials. Must be specified when `family` is `binomial`. |
| `family` | This is a family object specifying the distribution and link to use in fitting etc. See `glm` and `family` for more details. Currently support Poisson and binomial distributions. |
| `data` | A data frame or list containing the model response variable and covariates required by the formula. |
| `...` | Additional arguments to be passed to the low level regression fitting functions. |

## Details

It is necessary to assess whether there is zero-inflation in count data, e.g., Poisson or binomial data. The model selection approach can be used to determine whether a zero-inflated model is needed. To do that, we can fit both a ZIGAM and a regular GAM to the data, and then compare the logarithmic marginal likelihoods from these two models. Higher logarithmic marginal likelihood from the ZIGAM would indicate that there is zero-inflation in the count data. Otherwise, we can simply fit a regular GAM instead of a ZIGAM. See Liu and Chan (2010) for more detail.

## Value

A list containing the following components:

| | |
|---|---|
| `fit.gam` | A fitted GAM assuming there is no zero-inflation in the data. |
| `V.beta` | The estimated covariance matrix of the GAM. |
| `mu` | The fitted mean values. |
| `formula` | Model formula. |
| `family` | The family used. |
| `loglik, ploglik` | The (penalized) log-likelihood of the fitted model. |
| `logE` | Approximated logarithmic marginal likelihood by Laplace method used for model selection. |

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

zigam

## Examples

```
## Poisson Response
set.seed(11)
n <- 200
x1 <- runif(n, 0, 1)

eta0 <- f0(x1)/4 - 0.5
mu0 <- exp(eta0)

y <- rpois(rep(1,n), mu0) # generating non-zero-inflated data

res.gam <- disgam(y~s(x1), family=poisson) # fit a regular GAM
res.zigam <- zigam(y~s(x1), maxiter=10, family=poisson) # fit a ZIGAM

res.gam$logE > res.zigam$logE # compare the model selction criterion

# Another example
set.seed(11)
n <- 200
x1 <- runif(n, 0, 1)
eta0 <- f0(x1)/4 - 0.5
mu0 <- exp(eta0)

alpha0 <- 0.4
delta0 <- 0.8
p0 <- .Call("logit_linkinv", alpha0 + delta0 * eta0, PACKAGE = "stats")

# Generating zero-inflated Poisson count data
z <- rbinom(rep(1,n), 1, p0)
y <- rpois(rep(1,n), mu0)
y[z==0] <- 0

res.gam <- disgam(y~s(x1), family=poisson) # fit a regular GAM
res.zigam <- zigam(y~s(x1), family=poisson) # fit a ZIGAM

res.gam$logE < res.zigam$logE # compare the model selction criterion
```

---

eggdata *Pollock Egg Data*

---

## Description

This data set gives Pollock egg count data from cruise studies off the tip of Alaska in year of 1987.

## Usage

```
data(eggdata)
```

## Format

A data frame with 274 observations on the following 6 variables.

`bottom` A numeric vector: bottom depth at the sampling spot

`lat` A numeric vector: latitude of the sampling spot

`lon` A numeric vector: longitude of the sampling spot

`catch` A numeric vector: the number of eggs caught in the unit of CPUE (Capture Per Unit Effort) and “0" indicates a zero catch

`j.day` Julian day

`year` Year: 1987

## Author(s)

Hai Liu and Kung-Sik Chan

---

`PCOZIGAM.cts` *Fitting Proportionally Constrained Zero-Inflated Generalized Additive Models with Continuous Responses*

---

## Description

`PCOZIGAM.cts` fits a COZIGAM with proportional constraint for continuous exponential family responses.

`PCOZIGAM.dis` fits a COZIGAM with proportional constraint for discrete exponential family responses. One of the two funcions is called from `cozigam` automatically depending on the distribution family if the argument `constraint="proportional"`.

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

`cozigam`

PCOZIGAM.dis *Fitting Proportionally Constrained Zero-Inflated Generalized Additive Models with Discrete Responses*

## Description

PCOZIGAM.dis fits a COZIGAM with proportional constraint for discrete zero-inflated exponential family responses, e.g. Poisson and binomial distributions. If the regular exponential family admits 0 as a possible realization with positive probability, the EM algorithm is used in the estimation procedure.

PCOZIGAM.cts fits a COZIGAM with proportional constraint for continuous zero-inflated exponential family responses. One of the two funcions is called from cozigam automatically depending on the distribution family if the argument constraint="proportional".

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

cozigam

plot.cozigam *Default COZIGAM plotting*

## Description

Takes a fitted cozigam object produced by cozigam() and plots the component smooth functions that make it up, on the scale of the linear predictor.

## Usage

```
## S3 method for class 'cozigam'
plot(x, plot.2d = "contour", too.far = 0,
   n.1d = 100, n.2d = 30, theta = 30, phi = 30, select = NULL, image.col = "topo",
    persp.col = "lightblue", contour.col = "red", n.Col = 100, shade.ci = FALSE,
    shade.col = "gray80", Rug = TRUE, xlab, ylab, ...)
```

## Arguments

| | |
|---|---|
| `x` | A fitted `cozigam` object produced by `cozigam()`. |
| `plot.2d` | One of “contour" (default) or “persp". |
| `select` | Allows the plot for a single model term to be selected for printing. e.g. if you just want the plot for the second smooth term set `select=2`. |
| `n.1d` | Number of points used for each 1-D plot. Default value 100. |
| `n.2d` | Square root of number of points used to grid estimates of 2-D functions for contouring. |
| `theta` | One of the perspective plot angles. |
| `phi` | The other perspective plot angle. |
| `too.far` | If greater than 0 then this is used to determine when a location is too far from data to be plotted when plotting 2-D smooths. This is useful since smooths tend to go wild away from data. The data are scaled into the unit square before deciding what to exclude, and `too.far` is a distance within the unit square. |
| `shade.ci` | Logical. If `TRUE`, produce shaded regions as confidence bands for smooths. |
| `shade.col` | Define the color used for shading confidence bands. |
| `image.col` | Define the color used for 2-D image plots. |
| `persp.col` | Define the color used for 2-D perspective plots. |
| `contour.col` | Define the color used for the 2-D contour lines. |
| `n.Col` | Control the number of colors in 2-D image plots. |
| `Rug` | Logical, if `TRUE` (default) then the covariate to which the plot applies is displayed as a rug plot at the foot of each plot of a 1-D smooth, and the locations of the covariates are plotted as points on the contour plot representing a 2-D smooth. |
| `xlab, ylab` | Titles for the x and y axes. |
| `...` | Other graphics parameters to pass on to plotting commands. |

## Details

Produces default plot showing the smooth components of a fitted `cozigam`.

Smooths of more than 2 variables are not currently dealt with, but simply generate a warning.

## Value

The function simply generates plots.

## Author(s)

Hai Liu and Kung-Sik Chan

## See Also

`cozigam`, `predict.cozigam`

### Examples

```
set.seed(11)
n <- 400
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)

f <- test(x1,x2)*4-mean(test(x1,x2)*4) + f0(x3)/2-mean(f0(x3)/2)
sig <- 0.5
mu0 <- f + 3
y <- mu0 + rnorm(n, 0, sig)

alpha0 <- -2.2
delta0 <- 1.2
p0 <- .Call("logit_linkinv", alpha0 + delta0 * mu0, PACKAGE = "stats")
z <- rbinom(rep(1,n), 1, p0)
y[z==0] <- 0

res <- cozigam(y~s(x1,x2)+s(x3), constraint = "proportional", family = gaussian)

plot(res, plot.2d = "contour", image.col="topo") # contour plot
plot(res, plot.2d = "persp", select=1) # perspective plot
```

---

## predict.cozigam — *Prediction from fitted COZIGAM*

---

### Description

Takes a fitted `cozigam` object produced by `cozigam()` and produces predictions given a new set of values for the model covariates or the original values used for the model fit. Predictions can be accompanied by standard errors, based on the distribution of the model coefficients obtained by Louis' method.

### Usage

```
## S3 method for class 'cozigam'
predict(object, newdata, type="link", se.fit=FALSE, ...)
```

### Arguments

| | |
|---|---|
| `object` | A fitted `cozigam` object as produced by `cozigam()`. |
| `newdata` | A data frame containing the values of the model covariates at which predictions are required. If this is not provided then predictions corresponding to the original data are returned. If `newdata` is provided then it must contain all the variables needed for prediction. |
| `type` | When this has the value `"link"` (default) the linear predictor (possibly with associated standard errors) is returned. When `type="terms"` each component of the linear predictor is returned seperately (possibly with standard errors): this |

excludes any intercept. When type="response" predictions on the scale of the response are returned (possibly with approximate standard errors).

se.fit Logical. If TRUE (not default), standard error estimates are returned for each prediction.

... Other arguments.

## Details

The standard errors produced by predict.cozigam() are based on the covariance matrix of the parameters obtained by Louis' method in the fitted gam object.

## Value

If se.fit is TRUE then a 3 item list is returned with items (both arrays) fit, se.fit containing predictions and associated standard error estimates and p containing predictions of associated zero-inflation rates, otherwise a 2 item list without the array of standard error estimated is returned. The dimensions of the returned arrays depends on whether type is "terms" or not: if it is then the array is 2 dimensional with each term in the linear predictor separate, otherwise the array is 1 dimensional and contains the linear predictor/predicted values (or corresponding s.e.s). The linear predictor returned termwise will not include the intercept.

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

Louis, T. A. (1982) Finding the Observed Information Matrix when Using EM Algorithm. J. R. Statist. Soc. B, 44, 226-233

## See Also

cozigam, plot.cozigam

## Examples

```
set.seed(11)
n <- 400
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)

f <- test(x1,x2)*4-mean(test(x1,x2)*4) + f0(x3)/2-mean(f0(x3)/2)
sig <- 0.5
mu0 <- f + 3
y <- mu0 + rnorm(n, 0, sig)
```

```
alpha0 <- -2.2
delta0 <- 1.2
p0 <- .Call("logit_linkinv", alpha0 + delta0 * mu0, PACKAGE = "stats")
z <- rbinom(rep(1,n), 1, p0)
y[z==0] <- 0

res <- cozigam(y~s(x1,x2)+s(x3), constraint = "proportional", family = gaussian)

newdata <- data.frame(x1=c(0.5,0.8), x2=c(0.2,0.1), x3=c(0.3,0.7))
predict(res, newdata=newdata, se.fit=TRUE, type="response")
```

---

`print.cozigam` *Constrained Zero-Inflated Generalized Additive Model default print statement*

---

## Description

This is the default print statement for a COZIGAM object.

## Author(s)

Hai Liu and Kung-Sik Chan

## See Also

`cozigam`, `summary.cozigam`

---

`summary.cozigam` *Summary for a COZIGAM fit*

---

## Description

Takes a fitted `cozigam` object produced by `cozigam()` and produces various useful summaries from it.

## Usage

```
## S3 method for class 'cozigam'
summary(object, dispersion = NULL, ...)

## S3 method for class 'summary.cozigam'
print(x,digits = max(3, getOption("digits") - 3),
                  signif.stars = getOption("show.signif.stars"), ...)
```

### Arguments

- `object` — a fitted `cozigam` object as produced by `cozigam()`.
- `dispersion` — a known dispersion parameter. `NULL` to use estimate or default (e.g. 1 for Poisson).
- `x` — a `summary.cozigam` object produced by `summary.cozigam()`.
- `digits` — the number of significant digits to use when printing.
- `signif.stars` — logical. If `TRUE`, “significance stars" are printed for each coefficient.
- `...` — other arguments.

### Details

`print.summary.cozigam` tries to be smart about formatting the coefficients, standard errors, etc. and additionally gives “significance stars" if signif.stars is `TRUE`.

### Value

`summary.cozigam` produces a list of summary information for a fitted `cozigam` object.

- `p.coeff` — an array of estimates of the strictly parametric model coefficients, including the linear constraints parameters.
- `p.t` — an array of the `p.coeff`'s divided by their standard errors.
- `p.pv` — an array of p-values for the null hypothesis that the corresponding parameter is zero. Calculated with reference to the t distribution with the estimated residual degrees of freedom for the model fit if the dispersion parameter has been estimated, and the standard normal if not.
- `m` — the number of smooth terms in the model.
- `chi.sq` — an array of test statistics for assessing the significance of model smooth terms. If $b_i$ is the parameter vector for the `i`th smooth term, and this term has estimated covariance matrix $V_i$ then the statistic is $b_i' V_i^{k-} b_i$, where $V_i^{k-}$ is the rank `k` pseudo-inverse of $V_i$, and `k` is estimated rank of $V_i$.
- `s.pv` — an array of approximate p-values for the null hypotheses that each smooth term is zero. Be warned, these are only approximate. In the case of known dispersion parameter, they are obtained by comparing the chi.sq statistic given above to the chi-squared distribution with `k` degrees of freedom, where `k` is the estimated rank of $V_i$. If the dispersion parameter is unknown (in which case it will have been estimated) the statistic is compared to an F distribution with `k` upper d.f. and lower d.f. given by the residual degrees of freedom for the model.Typically the p-values will be somewhat too low, because they are conditional on the smoothing parameters, which are usually uncertain, but note that the statistic can also have low power if the rank, `k`, is too high relative to the EDF of the term.
- `se` — array of standard error estimates for all parameter estimates.
- `edf` — array of estimated degrees of freedom for the model terms.
- `residual.df` — estimated residual degrees of freedom.
- `n` — number of data.

| | |
|---|---|
| `family` | the family used. |
| `formula` | the original GAM formula. |
| `dispersion` | estimated (or given) scale parameter. |
| `cov.unscaled` | the estimated covariance matrix of the parameters, divided by scale parameter. |
| `cov.scaled` | the estimated covariance matrix of the parameters. |
| `p.table` | significance table for parameters. |
| `s.table` | significance table for smooths. |

## Author(s)

Hai Liu and Kung-Sik Chan

## See Also

`cozigam`, `predict.cozigam`

## Examples

```
set.seed(1)
n <- 600
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)
x3 <- runif(n, 0, 1)

f <- test(x1, x2)*2 + f0(x3)/5
eta0 <- f/1.1
mu0 <- exp(eta0)

eta.p10 <- (test(x1,x2) - mean(test(x1,x2)))*2/1.1
eta.p20 <- (f0(x3) - mean(f0(x3)))/5/1.1

alpha0 <- 0.5
delta10 <- 1
delta20 <- 0
eta.p0 <- delta10*eta.p10 + delta20*eta.p20
p0 <- .Call("logit_linkinv", alpha0 + eta.p0, PACKAGE = "stats")

z <- rbinom(rep(1,n), 1, p0)
y <- rpois(rep(1,n), mu0)
y[z==0] <- 0; rm(z)

res <- cozigam(y~s(x1,x2)+s(x3), constraint="component", zero.delta=c(NA, 0), family=poisson)
summary(res)
```

test *Some Test Functions used in Simulation Study*

## Description

test defines a 2-D test function.

f0 and f1 define two 1-D test functions. The three test functions will be used in the simulation study of COZIGAM fits.

## Author(s)

Hai Liu and Kung-Sik Chan

## See Also

cozigam

zigam *Fitting (Unconstrained) Zero-Inflated Generalized Additive Models*

## Description

Fit a Zero-Inflated Generalized Additive Model (ZIGAM) to data.

## Usage

```
zigam (formula, maxiter = 20, conv.crit = 1e-3,
    size = NULL, log.tran = FALSE, family, data=list(), ...)
```

## Arguments

| | |
|---|---|
| formula | A GAM formula. This is exactly like the formula for a GLM except that smooth terms can be added to the right hand side of the formula (and a formula of the form y ~ . is not allowed). Smooth terms are specified by expressions of the form: s(var1,var2,...,k=12,fx=FALSE,bs="tp",by=a.var) where var1, var2, etc. are the covariates which the smooth is a function of and k is the dimension of the basis used to represent the smooth term. by can be used to specify a variable by which the smooth should be multiplied. |
| maxiter | The maximum number of iterations allowed in the EM algorithm in estimating discrete ZIGAMs. |
| conv.crit | Convergence criterion in the iterative estimation algorithm. |
| size | Number of trials. Must be specified when family is binomial. |
| log.tran | Logical. TRUE if log-transformation is needed for the response. |

| | |
|---|---|
| `family` | This is a family object specifying the distribution and link to use in fitting etc. See `glm` and `family` for more details. Currently support Gaussian/lognormal, Gamma, Poisson and binomial distributions. |
| `data` | A data frame or list containing the model response variable and covariates required by the formula. |
| `...` | Additional arguments to be passed to the low level regression fitting functions. |

## Details

A Zero-Inflated Generalized Additive Model (ZIGAM) assumes the response variable $Y_i$ is distributed from a zero-inflated 1-parameter exponential family with covariate $x_i$ (could be high dimensional). More specifically, $Y_i$ comes from a non-zero-inflated exponential family distribution $f(x_i)$ (regular component) with probability $p_i$ and equals zero with probability $1-p_i$. The probability of non-zero-inflation $p_i$ also depends on the covariates through some unknown smooth funtions. Different from the COnstrained Zero-Inflated Generalized Additive Model (COZIGAM), the process of generating the non-zero-inflated responses and the zero-inflation process are assumed to be independent. The mean of the non-zero-inflated exponential family distribution is assumed to be

$$g(\mu_i) = s_1(x_i),$$

and the non-zero-inflation probability is linked to the covariates by

$$logit(p_i) = s_2(x_i),$$

where $s_1$ and $s_2$ are two possibly distinct smooth functions to be estimated nonparametrically by Generalized Additive Models. See Liu and Chan (2010) for more detail.

## Value

A list containing the following components:

| | |
|---|---|
| `fit.gam` | A fitted GAM of the regular component, i.e., non-zero-inflated exponential family regression model. |
| `fit.lr` | A logistic regression model on the zero-inflation process. |
| `V.beta` | The covariance matrix of the estimated parameters associated with the smooth functions in the non-zero-inflated data generating process. |
| `V.gamma` | The covariance matrix of the estimated parameters associated with the smooth functions in the zero-inflation process. |
| `mu` | The fitted regular mean values. |
| `dispersion` | (Estimated) dispersion parameter. |
| `formula` | Model formula. |
| `p` | The fitted non-zero-inflation probabilities. |
| `psi` | Conditional expectation of the zero-inflation indicator (only for the discrete case which involves EM algorithm). |
| `family` | The family used. |
| `loglik, ploglik` | The (penalized) log-likelihood of the fitted model. |

| | |
|---|---|
| logE | Approximated logarithmic marginal likelihood by Laplace method used for model selection. |
| X1 | Design matrix in the non-zero-inflated exponential family regression model. |
| X2 | Design matrix in the logistic regression model. |

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

cozigam

## Examples

```
## Gaussian Response
set.seed(11)
n <- 200
x1 <- runif(n, 0, 1)
f <- (f0(x1)-mean(f0(x1)))/2
sig <- 0.3
mu0 <- f + 1.5
y <- mu0 + rnorm(n, 0, sig)

eta.p0 <- f1(x1)*2 - 1 # true function used in zero-infation process
p0 <- .Call("logit_linkinv", eta.p0, PACKAGE = "stats")

z <- rbinom(rep(1,n), 1, p0)
y[z==0] <- 0

# Fit a ZIGAM
res.un <- zigam(y~s(x1), family=gaussian)

# Compare with a COZIGAM
res <- cozigam(y~s(x1), family=gaussian)
res.un$logE > res$logE

## Poisson Response
set.seed(11)
n <- 400
x1 <- runif(n, 0, 1)
x2 <- runif(n, 0, 1)

eta0 <- test(x1,x2)*3
mu0 <- exp(eta0)
```

```
alpha0 <- -0.2
delta0 <- 1.0
p0 <- .Call("logit_linkinv", alpha0 + delta0 * eta0, PACKAGE = "stats")


z <- rbinom(rep(1,n), 1, p0)
y <- rpois(rep(1,n), mu0)
y[z==0] <- 0

res.un <- zigam(y~s(x1,x2), maxiter=30, family=poisson) # fit a ZIGAM
res <- cozigam(y~s(x1,x2), maxiter=30, family=poisson) # fit a COZIGAM
res.un$logE < res$logE # compare the model selction criterion
```

---

ZIGAM.cts *Fitting Continuous Zero-Inflated Generalized Additive Models*

---

## Description

ZIGAM.cts fits a ZIGAM for continuous zero-inflated exponential family responses.

ZIGAM.dis fits a ZIGAM for discrete zero-inflated exponential family responses. One of the two funcions is called from zigam automatically depending on the distribution family.

## Author(s)

Hai Liu and Kung-Sik Chan

## References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

## See Also

zigam

## ZIGAM.dis — *Fitting Discrete Zero-Inflated Generalized Additive Models*

### Description

`ZIGAM.dis` fits a ZIGAM for discrete zero-inflated exponential family responses.

`ZIGAM.cts` fits a ZIGAM for continuous zero-inflated exponential family responses. One of the two funcions is called from `zigam` automatically depending on the distribution family.

### Author(s)

Hai Liu and Kung-Sik Chan

### References

Liu, H and Chan, K.S. (2010) Introducing COZIGAM: An R Package for Unconstrained and Constrained Zero-Inflated Generalized Additive Model Analysis. Journal of Statistical Software, 35(11), 1-26. http://www.jstatsoft.org/v35/i11/

### See Also

`zigam`

# Index