

Package ‘gllvm’

January 2, 2019

Type Package

Title Generalized Linear Latent Variable Models

Version 1.1.2

Author

Jenni Niku, Wesley Brooks, Riki Herliansyah, Francis K.C. Hui, Sara Taskinen, David I. Warton

Maintainer Jenni Niku <jenni.m.e.niku@jyu.fi>

Description Analysis of multivariate data using generalized linear latent variable models (gllvm). Estimation is performed using either Laplace approximation method or variational approximation method implemented via TMB (Kristensen et al., (2016), <doi:10.18637/jss.v070.i05>). Details for gllvm, see Hui et al. (2015) <doi:10.1111/2041-210X.12236> and (2017) <doi:10.1080/10618600.2016.1164708> and Niku et al. (2017) <doi:10.1007/s13253-017-0304-7>.

License GPL-2

Imports MASS, Matrix, mvtnorm, statmod, fishMod

Depends TMB, mvabund

Encoding UTF-8

LazyData true

LinkingTo TMB, RcppEigen

RoxygenNote 6.1.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2019-01-02 13:10:16 UTC

R topics documented:

anova.gllvm	2
coefplot.gllvm	3
confint.gllvm	4
getResidualCor.gllvm	5
gllvm	6
logLik.gllvm	11

ordiplot.gllvm	12
plot.gllvm	13
residuals.gllvm	15
summary.gllvm	16

Index	18
--------------	-----------

anova.gllvm	<i>Analysis Of Deviance for gllvm</i>
-------------	---------------------------------------

Description

Computes an analysis of deviance table for two generalized linear latent variable model fits.

Usage

```
## S3 method for class 'gllvm'
anova(object, ...)
```

Arguments

object	an object of class 'gllvm'.
...	one or more objects of class 'gllvm'

Details

Computes likelihood-ratio test for two or more gllvm models. Test results makes sense only for nested models. Notice also that this test is not designed for testing models which have degrees of freedom difference larger than 20. For such models the P-value should be treated as very approximate.

Author(s)

Jenni Niku

Examples

```
## Load a dataset from the mvabund package
data(antTraits)
y <- antTraits$abund
X <- antTraits$env
TR <- antTraits$traits
# Fit gllvm model
fit1 <- gllvm(y, X, TR, formula = ~ Bare.cover + Shrub.cover, family = poisson())
fit2 <- gllvm(y, X, TR, formula = ~ Bare.cover +
              (Bare.cover + Shrub.cover) : Webers.length, family = poisson())
# Test if the model with fourth corner interaction terms is significantly
# better using likelihood-ratio test:
anova(fit1, fit2)
```

coefplot.gllvm

*Plot covariate coefficients and confidence intervals***Description**

Plots covariate coefficients and their confidence intervals.

Usage

```
## S3 method for class 'gllvm'
coefplot(object, y.label = TRUE, which.Xcoef = NULL,
         cex.ylab = 0.5, mfrow = NULL, mar = c(4, 6, 2, 1),
         xlim.list = NULL, ...)
```

Arguments

object	an object of class 'gllvm'.
y.label	logical, if TRUE (default) colnames of y with respect to coefficients are added to plot.
which.Xcoef	vector indicating which X-coefficients will be plotted. Can be vector of covariate names or numbers. Default is NULL when all covariate coefficients are plotted.
cex.ylab	the magnification to be used for axis annotation relative to the current setting of cex.
mfrow	same as mfrow in par. If NULL (default) it is determined automatically.
mar	vector of length 4, which defines the margin sizes: c(bottom, left, top, right). Defaults to c(4, 5, 2, 1).
xlim.list	list of vectors with length of two to define the intervals for x axis in each covariate plot. Defaults to NULL when the interval is defined by the range of point estimates and confidence intervals
...	additional graphical arguments.

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>, Francis K.C. Hui, Sara Taskinen

Examples

```
## Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
X <- as.matrix(antTraits$env)
# Fit model with environmental covariates
fit <- gllvm(y, X, formula = ~ Bare.ground + Shrub.cover,
            family = poisson())
coefplot.gllvm(fit)
```

```
# Fit model with all environmental covariates
fitx <- gllvm(y, X, family = "negative.binomial")
coefplot(fitx, mfrow = c(3,2))
coefplot(fitx, which.Xcoef = 1:2)

# Fit gllvm model with environmental and trait covariates
TR <- antTraits$traits
fitT <- gllvm(y = y, X = X, TR = TR, family = "negative.binomial")
coefplot(fitT)
```

confint.gllvm

Confidence intervals for model parameters

Description

Computes confidence intervals for parameters in a fitted gllvm model.

Usage

```
## S3 method for class 'gllvm'
confint(object, parm = NULL, level = 0.95, ...)
```

Arguments

object	an object of class 'gllvm'.
parm	a specification of which parameters are to be given confidence intervals, a vector of names. If missing, all parameters are considered.
level	the confidence level. Scalar between 0 and 1.
...	not used.

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>

Examples

```
## Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
X <- as.matrix(antTraits$env[,1:2])
# Fit gllvm model
fit <- gllvm(y = y, X = X, family = poisson())
# 95 % confidence intervals for coefficients of X variables
confint(fit, level = 0.95, parm = "Xcoef")
```

getResidualCor.gllvm *Extract residual correlations from gllvm object*

Description

Calculates the residual covariance and correlation matrices for gllvm model.

Usage

```
## S3 method for class 'gllvm'  
getResidualCor(object)
```

Arguments

object an object of class 'gllvm'.

Value

Function returns following components:

cor	residual correlation matrix
cov	residual covariance matrix
trace	trace of residual covariance matrix

Author(s)

Francis K.C. Hui, Jenni Niku, David I. Warton

Examples

```
# Load a dataset from the mvabund package  
data(antTraits)  
y <- as.matrix(antTraits$abund)  
# Fit gllvm model  
fit <- gllvm(y = y, family = poisson())  
# residual correlations:  
cr <- getResidualCor(fit)  
  
## Not run:  
# Plot residual correlations:  
install.packages("corrplot", "gclus")  
library(corrplot)  
library(gclus)  
rbPal <- colorRampPalette(c('darkblue', 'white', 'darkred'))  
breaks <- seq(min(cr$cor), max(cr$cor), length.out = 40)  
Colors <- rbPal(100)[as.numeric(cut(cr$cor, breaks = breaks))]  
corrplot(cr$cor[order.single(cr$cor), order.single(cr$cor)], diag = F,  
          type = "lower", method = "square", tl.cex = 0.8, tl.srt = 45, tl.col = "red")
```

```
## End(Not run)
```

gllvm

Generalized Linear Latent Variable Models

Description

Fits generalized linear latent variable model for multivariate data. The model can be fitted using Laplace approximation method or variational approximation method.

Usage

```
gllvm(y = NULL, X = NULL, TR = NULL, data = NULL, formula = NULL,
      num.lv = 2, family, method = "VA", row.eff = FALSE,
      offset = NULL, sd.errors = TRUE, Lambda.struc = "unstructured",
      diag.iter = 5, trace = FALSE, plot = FALSE,
      la.link.bin = "probit", n.init = 1, Power = 1.5, reltol = 1e-08,
      seed = NULL, max.iter = 200, maxit = 1000, start.fit = NULL,
      starting.val = "res", TMB = TRUE, optimizer = "optim",
      Lambda.start = c(0.1, 0.5), jitter.var = 0)
```

Arguments

y	(n x m) matrix of responses.
X	matrix or data.frame of environmental covariates.
TR	matrix or data.frame of trait covariates.
data	data in long format, that is, matrix of responses, environmental and trait covariates and row index named as 'id'. When used, model needs to be defined using formula. This is alternative data input for y, X and TR.
formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
num.lv	number of latent variables, d, in gllvm model. Non-negative integer, less than number of response variables (m). Defaults to 2.
family	distribution function for responses. Options are poisson(link = "log"), "negative.binomial" (with log link), binomial(link = "probit") (and also binomial(link = "logit") when method = "LA"), zero inflated poisson ("ZIP") and Tweedie ("tweedie") (with log link, only with "LA"-method), "ordinal" (only with "VA"-method).
method	model can be fitted using Laplace approximation method (method = "LA") or variational approximation method (method = "VA"). Defaults to "VA".
row.eff	FALSE, TRUE or "random", Indicating whether row effects are included in the model as a fixed or as a random effects. Defaults to FALSE when row effects are not included.

offset	vector or matrix of offset terms.
sd.errors	logical. If TRUE (default) standard errors for parameter estimates are calculated.
Lambda.struc	covariance structure of VA distributions for latent variables when method = "VA", "unstructured" or "diagonal".
diag.iter	non-negative integer which is used to speed up the updating of variational (covariance) parameters in VA method. Defaults to 5.
trace	logical, if TRUE in each iteration step information on current step will be printed. Defaults to FALSE. Only with TMB = FALSE.
plot	logical, if TRUE ordination plots will be printed in each iteration step when TMB = FALSE. Defaults to FALSE.
la.link.bin	link function for binomial family if method = "LA". Options are "logit" and "probit".
n.init	number of initial runs. Uses multiple runs and picks up the one giving highest log-likelihood value. Defaults to 1.
Power	fixed power parameter in Tweedie model. Scalar from interval (1,2). Defaults to 1.5.
reltol	convergence criteria for log-likelihood, defaults to 1e-6.
seed	a single seed value, defaults to NULL.
max.iter	maximum number of iterations when TMB = FALSE, defaults to 200.
maxit	maximum number of iterations within optim function, defaults to 1000.
start.fit	object of class 'gllvm' which can be given as starting parameters for count data (poisson, NB, or ZIP).
starting.val	starting values can be generated by fitting model without latent variables, and applying factorial analysis to residuals to get starting values for latent variables and their coefficients (starting.val = "res"). Another options are to use zeros as a starting values (starting.val = "zero") or initialize starting values for latent variables with (n x num.lv) matrix. Defaults to "res", which is recommended.
TMB	logical, if TRUE model will be fitted using Template Model Builder (TMB). TMB is always used if method = "LA". Defaults to TRUE.
optimizer	if TMB=TRUE, log-likelihood can be optimized using "optim" (default) or "nlminb".
Lambda.start	starting values for variances in VA distributions for latent variables in variational approximation method. Defaults to 0.1.
jitter.var	jitter variance for starting values of latent variables. Defaults to 0, meaning no jittering.

Details

Fits generalized linear latent variable models as in Hui et al. (2015 and 2017) and Niku et al. (2017). Method can be used with two types of latent variable models depending on covariates. If only site related environmental covariates are used, the expectation of response Y_{ij} is determined by

$$g(\mu_{ij}) = \eta_{ij} = \alpha_i + \beta_{0j} + x'_i \beta_j + u'_i \theta_j,$$

where $g(\cdot)$ is a known link function, u_i are d -variate latent variables ($d \ll m$), α_i is an optional row effect at site i , and it can be fixed or random effect, β_{0j} is an intercept term for species j , β_j and θ_j are column specific coefficients related to covariates and the latent variables, respectively.

An alternative model is the fourth corner model (Brown et al., 2014, Warton et al., 2015) which will be fitted if also trait covariates are included. The expectation of response Y_{ij} is

$$g(\mu_{ij}) = \alpha_i + \beta_{0j} + x_i' \beta_x + TR_j' \beta_t + \text{vec}(B) * \text{kronecker}(TR_j, X_i) + u_i' \theta_j$$

where $g(\cdot)$, u_i , β_{0j} and θ_j are defined as above. Vectors β_x and β_t are the main effects or coefficients related to environmental and trait covariates, respectively, matrix B includes interaction terms. The interaction/fourth corner terms are optional as well as are the main effects of trait covariates.

The method is sensitive for the choices of initial values of the latent variables. Therefore it is recommendable to use multiple runs and pick up the one giving the highest log-likelihood value. However, sometimes this is computationally too demanding, and default option `starting.val = "res"` is recommended. For more details on different starting value methods, see Niku et al., (2018).

Models are implemented using TMB (Kristensen et al., 2015) applied to variational approximation (Hui et al., 2017) and Laplace approximation (Niku et al., 2017).

An exception is ordinal family which is not implemented with TMB and therefore also `row.eff = "random"` does not work. With ordinal family response classes must start from 0 or 1.

Distributions:

Mean and variance for distributions are defined as follows.

- For count data family = `poisson()`: Expectation $E[Y_{ij}] = \mu_{ij}$, variance $V(\mu_{ij}) = \mu_{ij}$, or
- family = "negative.binomial": Expectation $E[Y_{ij}] = \mu_{ij}$, variance $V(\mu_{ij}) = \mu_{ij} + \phi_j * \mu_{ij}^2$, or
- family = "ZIP": Expectation $E[Y_{ij}] = (1-p)\mu_{ij}$, variance $V(\mu_{ij}) = \mu_{ij}(1-p)(1+\mu_{ij}p)$.
- For binary data family = `binomial()`: Expectation $E[Y_{ij}] = \mu_{ij}$, variance $V(\mu_{ij}) = \mu_{ij}(1 - \mu_{ij})$.
- For biomass data family = "tweedie": Expectation $E[Y_{ij}] = \mu_{ij}$, variance $V(\mu_{ij}) = \phi_j * \mu_{ij}^\nu$, where ν is a power parameter of Tweedie distribution. See details Dunn and Smyth (2005).
- For ordinal data family = "ordinal": Cumulative probit model, see Hui et al. (2016).

Value

An object of class "gllvm" includes the following components:

<code>call</code>	function call
<code>logL</code>	log likelihood
<code>lvs</code>	latent variables
<code>params</code>	list of parameters <ul style="list-style-type: none"> • <code>\$theta</code> coefficients related to latent variables • <code>\$beta0</code> column specific intercepts • <code>\$Xcoef</code> coefficients related to environmental covariates X

- \$B coefficients in fourth corner model
- \$row.params row-specific intercepts
- \$phi dispersion parameters ϕ for negative binomial or Tweedie family, or probability of zero inflation for ZIP family
- \$inv.phi dispersion parameters $1/\phi$ for negative binomial

Power	power parameter ν for Tweedie family
sd	list of standard errors of parameters
prediction.errors	list of prediction covariances for latent variables and variances for random row effects when method "LA" is used
A, Ar	covariance matrices for variational densities of latent variables and variances for random row effects

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>, Wesley Brooks, Riki Herliansyah, Francis K.C. Hui, Sara Taskinen, David I. Warton

References

- Brown, A. M., Warton, D. I., Andrew, N. R., Binns, M., Cassis, G., and Gibb, H. (2014). The fourth-corner solution - using predictive models to understand how species traits interact with the environment. *Methods in Ecology and Evolution*, 5:344-352.
- Dunn, P. K. and Smyth, G. K. (2005). Series evaluation of tweedie exponential dispersion model densities. *Statistics and Computing*, 15:267-280.
- Hui, F. K. C., Taskinen, S., Pledger, S., Foster, S. D., and Warton, D. I. (2015). Model-based approaches to unconstrained ordination. *Methods in Ecology and Evolution*, 6:399-411.
- Hui, F. K. C., Warton, D., Ormerod, J., Haapaniemi, V., and Taskinen, S. (2017). Variational approximations for generalized linear latent variable models. *Journal of Computational and Graphical Statistics*. *Journal of Computational and Graphical Statistics*, 26:35-43.
- Kasper Kristensen, Anders Nielsen, Casper W. Berg, Hans Skaug, Bradley M. Bell (2016). TMB: Automatic Differentiation and Laplace Approximation. *Journal of Statistical Software*, 70(5), 1-21.
- Niku, J., Warton, D. I., Hui, F. K. C., and Taskinen, S. (2017). Generalized linear latent variable models for multivariate count and biomass data in ecology. *Journal of Agricultural, Biological, and Environmental Statistics*, 22:498-522
- Niku, J., Brooks, W., Herliansyah, R., Hui, F. K. C., Taskinen, S., and Warton, D. I. (2018). Efficient estimation of generalized linear latent variable models. Submitted.
- Warton, D. I., Guillaume Blanchet, F., O'Hara, R. B., Ovaskainen, O., Taskinen, S., Walker, S. C. and Hui, F. K. C. (2015). So many variables: Joint modeling in community ecology. *Trends in Ecology & Evolution*, 30:766-779.

See Also

[coefplot.gllvm](#), [confint.gllvm](#), [ordiplot.gllvm](#), [plot.gllvm](#), [residuals.gllvm](#), [summary.gllvm](#).

Examples

```

## Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
X <- as.matrix(antTraits$env)
TR <- antTraits$traits
# Fit model with environmental covariates Bare.ground and Shrub.cover
fit <- gllvm(y, X, formula = ~ Bare.ground + Shrub.cover,
            family = poisson())
ordiplot(fit)
coefplot(fit)

## Example 1: Fit model with two latent variables
# Using variational approximation:
fitv0 <- gllvm(y, family = "negative.binomial", method = "VA")
ordiplot(fitv0)
plot(fitv0, mfrow = c(2,2))
summary(fitv0)
confint(fitv0)
# Using Laplace approximation: (this line may take about 30 sec to run)
fitl0 <- gllvm(y, family = "negative.binomial", method = "LA")
ordiplot(fitl0)

# Poisson family:
fit.p <- gllvm(y, family = poisson(), method = "LA")
ordiplot(fit.p)
# Use poisson model as a starting parameters for ZIP-model, this line may take few minutes to run
fit.z <- gllvm(y, family = "ZIP", method = "LA", start.fit = fit.p)
ordiplot(fit.z)

## Example 2: gllvm with environmental variables
# Fit model with two latent variables and all environmental covariates,
fitvX <- gllvm(formula = y ~ X, family = "negative.binomial")
ordiplot(fitvX, biplot = TRUE)
coefplot(fitvX)
# Fit model with environmental covariates Bare.ground and Shrub.cover
fitvX2 <- gllvm(y, X, formula = ~ Bare.ground + Shrub.cover,
              family = "negative.binomial")
ordiplot(fitvX2)
coefplot(fitvX2)
# Use 5 initial runs and pick the best one
fitvX_5 <- gllvm(y, X, formula = ~ Bare.ground + Shrub.cover,
              family = "negative.binomial", n.init = 5, jitter.var = 0.1)
ordiplot(fitvX_5)
coefplot(fitvX_5)

## Example 3: Data in long format
# Reshape data to long format:
datalong <- reshape(data.frame(cbind(y,X)), direction = "long",
                  varying = colnames(y), v.names = "y")

```

```

head(datalong)
fitvLong <- gllvm(data = datalong, formula = y ~ Bare.ground + Shrub.cover,
  family = "negative.binomial")

## Example 4: Fourth corner model
# Fit fourth corner model with two latent variables
fitF1 <- gllvm(y = y, X = X, TR = TR, family = "negative.binomial")
coefplot(fitF1)
# Fourth corner can be plotted also with next lines
#fourth = fitF1$fourth.corner
#library(lattice)
#a = max( abs(fourth) )
#colort = colorRampPalette(c("blue","white","red"))
#plot.4th = levelplot(t(as.matrix(fourth)), xlab = "Environmental Variables",
#  ylab = "Species traits", col.regions = colort(100),
#  at = seq( -a, a, length = 100), scales = list( x = list(rot = 45)))
#print(plot.4th)

# Specify model using formula
fitF2 <- gllvm(y = y, X = X, TR = TR,
  formula = ~ Bare.ground + Canopy.cover * (Pilosity + Webers.length),
  family = "negative.binomial")
ordiplot(fitF2)
coefplot(fitF2)

## Example 5: Fit Tweedie model
# Load coral data
data(tikus)
ycoral <- tikus$abund
# Let's consider only years 1981 and 1983
ycoral <- ycoral[((tikus$x$time == 81) + (tikus$x$time == 83)) > 0, ]
# Exclude species which have observed at less than 4 sites
ycoral <- ycoral[-17, (colSums(ycoral > 0) > 3)]
# Fit Tweedie model for coral data (this line may take few minutes to run)
fit.twe <- gllvm(y = ycoral, family = "tweedie", method = "LA")
ordiplot(fit.twe)

## Example 6: Random row effects
fitRand <- gllvm(y, family = "negative.binomial", row.eff = "random")
ordiplot(fitRand, biplot = TRUE)

```

logLik.gllvm

Log-likelihood of gllvm

Description

Extracts Log-likelihood from 'gllvm' objects.

Usage

```
## S3 method for class 'gllvm'
logLik(object, ...)
```

Arguments

```
object      an object of class 'gllvm'.
...         not used.
```

Author(s)

David I. Warton, Jenni Niku

Examples

```
## Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
# Fit gllvm model
fit <- gllvm(y = y, family = poisson())
# log-Likelihood:
logLik(fit)
```

```
ordiplot.gllvm      Plot latent variables from gllvm model
```

Description

Plots latent variables and their corresponding coefficients (biplot).

Usage

```
## S3 method for class 'gllvm'
ordiplot(object, biplot = FALSE, ind.spp = NULL,
  alpha = 0.5, main = NULL, which.lvs = c(1, 2), jitter = FALSE,
  s.colors = 1, symbols = FALSE, cex.spp = 0.7, ...)
```

Arguments

```
object      an object of class 'gllvm'.
biplot      TRUE if both latent variables and their coefficients are plotted, FALSE if only
            latent variables.
ind.spp     the number of response variables (usually, species) to include on the biplot. The
            default is none, or all if biplot = TRUE.
alpha       a numeric scalar between 0 and 1 that is used to control the relative scaling of
            the latent variables and their coefficients, when constructing a biplot.
```

main	main title.
which.lvs	indices of two latent variables to be plotted if number of the latent variables is more than 2. A vector with length of two. Defaults to c(1, 2).
jitter	if TRUE, jittering is applied on points.
s.colors	colors for sites
symbols	logical, if TRUE sites are plotted using symbols, if FALSE (default) site numbers are used
cex.spp	size of species labels in biplot
...	additional graphical arguments.

Details

Function constructs a scatter plot of two latent variables, i.e. an ordination plot. If only one latent variable is in the fitted model, latent variables are plotted against their corresponding row indices. The latent variables are labeled using the row index of the response matrix y .

Coefficients related to latent variables are plotted in the same figure with the latent variables if `biplot = TRUE`. They are labeled using the column names of y . The number of latent variable coefficients to be plotted can be controlled by `ind.spp`. An argument `alpha` is used to control the relative scaling of the latent variables and their coefficients. If `alpha = 0.5`, the latent variables and their coefficients are on the same scale.

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>, Francis K.C. Hui

Examples

```
### Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
fit <- gllvm(y, family = poisson())
# Ordination plot:
ordiplot(fit)
# Biplot with 10 species
ordiplot(fit, biplot = TRUE, ind.spp = 10)
```

Description

Four plots (selectable by `which`) are currently available: a plot of residuals against linear predictors of fitted values, a Normal Q-Q plot of residuals, residuals against row index and residuals against column index.

Usage

```
## S3 method for class 'gllvm'
plot(x, which = 1:5,
     caption = c("Residuals vs linear predictors", "Normal Q-Q",
                 "Residuals vs row index", "Residuals vs column index", "Scale-Location"),
     var.colors = NULL, panel = if (add.smooth) panel.smooth else points,
     add.smooth = if (!is.null(getOption("add.smooth"))) {
       getOption("add.smooth") } else TRUE, envelopes = TRUE, reps = 150,
     ...)
```

Arguments

x	an object of class 'gllvm'.
which	if a subset of the plots is required, specify a subset of the numbers 1:5, see caption below.
caption	captions to appear above the plots.
var.colors	colors for responses, vector with length of number of response variables or 1. Defaults to NULL, when different responses have different colors.
panel	panel function
add.smooth	logical indicating if a smoother should be added to most plots; see also panel above.
envelopes	logical, indicating if simulated point-wise confidence interval envelope will be added to Q-Q plot, defaults to TRUE
reps	number of replications when simulating confidence envelopes for normal Q-Q plot
...	additional graphical arguments.

Details

plot.gllvm is used for model diagnostics. Dunn-Smyth residuals (randomized quantile residuals) (Dunn and Smyth, 1996) are used in plots. Colors indicate different species.

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>

References

Dunn, P. K., and Smyth, G. K. (1996). Randomized quantile residuals. *Journal of Computational and Graphical Statistics*, 5, 236-244.

Hui, F. K. C., Taskinen, S., Pledger, S., Foster, S. D., and Warton, D. I. (2015). Model-based approaches to unconstrained ordination. *Methods in Ecology and Evolution*, 6:399-411.

See Also

[gllvm](#), [residuals.gllvm](#)

Examples

```
## Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
# Fit gllvm model with Poisson family
fit <- gllvm(y, family = poisson())
# Plot residuals
plot(fit, mfrow = c(3,2))

# Fit gllvm model with negative binomial family
fitnb <- gllvm(y = y, family = "negative.binomial")
# Plot residuals
plot(fitnb, mfrow = c(3,2))
# Plot only two first plots
plot(fitnb, which = 1:2, mfrow = c(1,2))
```

residuals.gllvm

*Dunn-Smyth residuals for gllvm model***Description**

Calculates Dunn-Smyth residuals for gllvm model.

Usage

```
## S3 method for class 'gllvm'
residuals(object, ...)
```

Arguments

object	an object of class 'gllvm'.
...	not used.

Details

Computes Dunn-Smyth residuals (randomized quantile residuals, Dunn and Smyth, 1996) for gllvm model. For the observation Y_{ij} Dunn-Smyth residuals are defined as

$$r_{ij} = \Phi^{-1}(u_{ij}F_{ij}(y_{ij}) + (1 - u_{ij})F_{ij}^{-}(y_{ij})),$$

where $\Phi(\cdot)$ and $F_{ij}(\cdot)$ are the cumulative probability functions of the standard normal distribution, $F_{ij}^{-}(y)$ is the limit as $F_{ij}(y)$ is approached from the negative side, and u_{ij} has been generated at random from the standard uniform distribution.

Value

A list containing residuals which is a matrix of residuals and linpred which is a matrix of linear predictors.

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>

References

Dunn, P. K., and Smyth, G. K. (1996). Randomized quantile residuals. *Journal of Computational and Graphical Statistics*, 5, 236-244.

Hui, F. K. C., Taskinen, S., Pledger, S., Foster, S. D., and Warton, D. I. (2015). Model-based approaches to unconstrained ordination. *Methods in Ecology and Evolution*, 6:399-411.

Examples

```
# Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
# Fit gllvm model
fit <- gllvm(y = y, family = poisson())
# residuals
res <- residuals(fit)
```

summary.gllvm

Summarizing gllvm model fits

Description

A summary of the fitted 'gllvm' object, including function call, distribution family and model parameters.

Usage

```
## S3 method for class 'gllvm'
summary(object, ...)
```

Arguments

object	an object of class 'gllvm'
...	not used.

Author(s)

Jenni Niku <jenni.m.e.niku@jyu.fi>

Examples

```
## Load a dataset from the mvabund package
data(antTraits)
y <- as.matrix(antTraits$abund)
# Fit gllvm model
fit <- gllvm(y = y, family = poisson())
summary(fit)
```

Index

`anova.gllvm`, 2

`coefplot (coefplot.gllvm)`, 3

`coefplot.gllvm`, 3, 9

`confint.gllvm`, 4, 9

`getResidualCor (getResidualCor.gllvm)`, 5

`getResidualCor.gllvm`, 5

`gllvm`, 6, 14

`logLik.gllvm`, 11

`nlminb`, 7

`optim`, 7

`ordiplot (ordiplot.gllvm)`, 12

`ordiplot.gllvm`, 9, 12

`plot.gllvm`, 9, 13

`residuals.gllvm`, 9, 14, 15

`summary.gllvm`, 9, 16