

Package ‘mcglm’

October 13, 2022

Type Package

Title Multivariate Covariance Generalized Linear Models

Version 0.8.0

Date 2022-09-15

Author Wagner Hugo Bonat [aut, cre]

Maintainer Wagner Hugo Bonat <wbonat@ufpr.br>

Description Fitting multivariate covariance generalized linear models (McGLMs) to data. McGLM is a general framework for non-normal multivariate data analysis, designed to handle multivariate response variables, along with a wide range of temporal and spatial correlation structures defined in terms of a covariance link function combined with a matrix linear predictor involving known matrices. The models take non-normality into account in the conventional way by means of a variance function, and the mean structure is modelled by means of a link function and a linear predictor. The models are fitted using an efficient Newton scoring algorithm based on quasi-likelihood and Pearson estimating functions, using only second-moment assumptions. This provides a unified approach to a wide variety of different types of response variables and covariance structures, including multivariate extensions of repeated measures, time series, longitudinal, spatial and spatio-temporal structures. The package offers a user-friendly interface for fitting McGLMs similar to the `glm()` R function. See Bonat (2018) <[doi:10.18637/jss.v084.i04](https://doi.org/10.18637/jss.v084.i04)>, for more information and examples.

Depends R (>= 4.2.0)

Suggests testthat, knitr, rmarkdown, MASS, mvtnorm, tweedie, devtools

Imports stats, Matrix, assertthat, graphics, Rcpp (>= 0.12.16)

License GPL-3 | file LICENSE

LazyData TRUE

URL mcglm.leg.ufpr.br

BugReports <https://github.com/wbonat/mcglm/issues>

Encoding UTF-8

VignetteBuilder knitr

RoxygenNote 7.2.0

LinkingTo Rcpp, RcppArmadillo

NeedsCompilation yes

Repository CRAN

Date/Publication 2022-09-15 19:36:07 UTC

R topics documented:

ahs	3
anova.mcglm	4
coef.mcglm	5
confint.mcglm	6
ESS	7
fitted.mcglm	7
fit_mcglm	8
gof	10
GOSHO	11
Hunting	12
mcglm	13
mc_anova_disp	14
mc_bias_corrected_std	15
mc_car	16
mc_complete_data	17
mc_compute_rho	18
mc_conditional_test	19
mc_dglm	20
mc_dist	21
mc_id	22
mc_initial_values	23
mc_link_function	24
mc_ma	26
mc_matrix_linear_predictor	27
mc_mixed	28
mc_ns	29
mc_remove_na	30
mc_robust_std	31
mc_rw	32
mc_sic	33
mc_sic_covariance	34
mc_twin	35
mc_variance_function	37
NewBorn	38
pAIC	40
pBIC	40

pKLIC	41
plogLik	42
plot.mcglm	42
print.mcglm	43
residuals.mcglm	44
RJC	45
soil	46
soya	47
summary.mcglm	48
vcov.mcglm	49

Index	50
--------------	-----------

ahs

Australian Health Survey

Description

The Australian health survey was used by Bonat and Jorgensen (2016) as an example of multivariate count regression model. The data consists of five count response variables concerning health system access measures and nine covariates concerning social conditions in Australian for 1987-88.

- sex - Factor with levels male and female.
- age - Respondent's age in years divided by 100.
- income - Respondent's annual income in Australian dollars divided by 1000.
- levypplus - Coded factor. If respondent is covered by private health insurance fund for private patients in public hospital with doctor of choice (1) or otherwise (0).
- freepoor - Coded factor. If respondent is covered by government because low income, recent immigrant, unemployed (1) or otherwise (0).
- freerepa - Coded factor. If respondent is covered free by government because of old-age or disability pension, or because invalid veteran or family of deceased veteran (1) or otherwise (0).
- illnes - Number of illnesses in past 2 weeks, with 5 or illnesses coded as 5.
- actdays - Number of days of reduced activity in the past two weeks due to illness or injury.
- hscore - Respondent's general health questionnaire score using Goldberg's method. High score indicates poor health.
- chcond - Factor with three levels. If respondent has chronic condition(s) and is limited in activity (limited), or if the respondent has chronic condition(s) but is not limited in activity (nonlimited) or otherwise (otherwise, reference level).
- Ndoc - Number of consultations with a doctor or specialist (response variable).
- Nndoc - Number of consultations with health professionals (response variable).
- Nadm - Number of admissions to a hospital, psychiatric hospital, nursing or convalescence home in the past 12 months (response variable).
- Nhosp - Number of nights in a hospital during the most recent admission.
- Nmed - Total number of prescribed and non prescribed medications used in the past two days.

Usage

```
data(ahs)
```

Format

a data.frame with 5190 records and 15 variables.

Source

Deb, P. and Trivedi, P. K. (1997) Demand for medical care by the elderly: A finite mixture approach. *Journal of Applied Econometrics* 12(3):313–336.

Bonat, W. H. and Jorgensen, B. (2016) Multivariate covariance generalized linear models. *Journal of Royal Statistical Society - Series C* 65:649–675.

Examples

```
require(mcglm)
data(ahs, package="mcglm")
form1 <- Ndoc ~ income + age
form2 <- Nndoc ~ income + age
Z0 <- mc_id(ahs)
fit.ahs <- mcglm(linear_pred = c(form1, form2),
                 matrix_pred = list(Z0, Z0), link = c("log", "log"),
                 variance = c("poisson_tweedie", "poisson_tweedie"),
                 data = ahs)
summary(fit.ahs)
```

anova.mcglm

Anova Tables

Description

Performs Wald tests of the significance for the linear predictor components by response variables. This function is useful for joint hypothesis tests of regression coefficients associated with categorical covariates with more than two levels. It is not designed for model comparison.

Usage

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

Arguments

object an object of class mcglm, usually, a result of a call to mcglm() function.

... additional arguments affecting the summary produced. Note that there is no extra options for mcglm object class.

Value

A data.frame with Chi-square statistic to test the null hypothesis of a parameter, or a set of parameters, be zero. Degree of freedom (Df) and p-values. The Wald test based on the observed covariance matrix of the parameters is used.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Examples

```
x1 <- seq(-1, 1, l = 100)
x2 <- gl(5, 20)
beta <- c(5, 0, -2, -1, 1, 2)
X <- model.matrix(~ x1 + x2)
set.seed(123)
y <- rnorm(100, mean = X%%beta, sd = 1)
data = data.frame("y" = y, "x1" = x1, "x2" = x2)
fit.anova <- mcglm(c(y ~ x1 + x2), list(mc_id(data)), data = data)
anova(fit.anova)
```

coef.mcglm

Model Coefficients

Description

Extract model coefficients for objects of mcglm class.

Usage

```
## S3 method for class 'mcglm'
coef(
  object,
  std.error = FALSE,
  response = c(NA, 1:length(object$beta_names)),
  type = c("beta", "tau", "power", "correlation"),
  ...
)
```

Arguments

object	an object of mcglm class.
std.error	logical. If TRUE returns the standard errors for the estimates. Default is FALSE.
response	a numeric vector specifying for which response variable the coefficients should be returned.

type a string vector (can be 1 element length) specifying which coefficients should be returned.
Options are "beta", "tau", "power", "tau" and "correlation".

... additional arguments affecting the summary produced. Note that there is no extra options for mcglm object class.

Value

A data.frame with parameters names, estimates, response variable number and parameters type.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

confint.mcglm

Confidence Intervals for Model Parameters

Description

Computes confidence intervals for parameters in a fitted mcglm model.

Usage

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

Arguments

object a fitted mcglm object.

parm specifies for which parameters are to be given confidence intervals, either a vector of numbers or a vector of strings. If missing, all parameters are considered.

level the nominal confidence level.

... additional arguments affecting the confidence interval produced. Note that there is no extra options for mcglm object class.

Value

A data.frame with confidence intervals, parameters names, response variable number and parameters type.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

ESS	<i>Generalized Error Sum of Squares</i>
-----	---

Description

Extract the generalized error sum of squares (ESS) for objects of `mcglm` class.

Usage

```
ESS(object, verbose = TRUE)
```

Arguments

<code>object</code>	an object or a list of objects representing a model of <code>mcglm</code> class.
<code>verbose</code>	logical. Print or not the ESS value.

Value

Returns the value of the generalized error sum of squares (ESS).

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

Wang, M. (2014). Generalized Estimating Equations in Longitudinal Data Analysis: A Review and Recent Developments. *Advances in Statistics*, 1(1)1–13.

See Also

`gof`, `plogLik`, `pAIC`, `pKLIC`, `GOSH0` and `RJC`.

<code>fitted.mcglm</code>	<i>Fitted Values</i>
---------------------------	----------------------

Description

Extract fitted values for objects of `mcglm` class.

Usage

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

Arguments

object an object of mcglm class.
 ... additional arguments affecting the summary produced. Note that there is no extra options for mcglm object class.

Value

A matrix with fitted values.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

 fit_mcglm

Chaser and Reciprocal Likelihood Algorithms

Description

This function implements the two main algorithms used for fitting multivariate covariance generalized linear models. The chaser and the reciprocal likelihood algorithms.

Usage

```
fit_mcglm(list_initial, list_link, list_variance,
          list_covariance, list_X, list_Z, list_offset,
          list_Ntrial, list_power_fixed, list_sparse,
          y_vec, correct, max_iter, tol, method,
          tuning, verbose, weights)
```

Arguments

list_initial a list of initial values for regression and covariance parameters.
 list_link a list specifying the link function names.
 Options are: "logit", "probit", "cauchit", "cloglog", "loglog", "identity",
 "log", "sqrt", "1/mu^2" and "inverse".
 See [mc_link_function](#) for details. Default link = "identity".
 list_variance a list specifying the variance function names. Options are: "constant", "tweedie",
 "poisson_tweedie", "binomialP" and "binomialPQ". See [mc_variance_function](#)
 for details. Default variance = "constant".
 list_covariance a list of covariance function names. Options are: "identity", "inverse" and
 "expm". Default covariance = "identity".
 list_X a list of design matrices. See [model.matrix](#) for details.
 list_Z a list of known matrices to compose the matrix linear predictor.
 list_offset a list of offset values. Default NULL.

<code>list_Ntrial</code>	a list of number of trials, useful only when analysing binomial data. Default 1.
<code>list_power_fixed</code>	a list of logicals indicating if the power parameters should be estimated or not. Default <code>power_fixed = TRUE</code> .
<code>list_sparse</code>	a list of logicals indicating if the matrices should be set up as sparse matrices. This argument is useful only when using exponential-matrix covariance link function. In the other cases the algorithm detects automatically if the matrix should be sparse or not.
<code>y_vec</code>	a vector of the stacked response variables.
<code>correct</code>	a logical indicating if the algorithm will use the correction term or not. Default <code>correct = FALSE</code> .
<code>max_iter</code>	maximum number of iterations. Default <code>max_iter = 20</code> .
<code>tol</code>	a numeric specifying the tolerance. Default <code>tol = 1e-04</code> .
<code>method</code>	a string specifying the method used to fit the models (" <code>chaser</code> " or " <code>rc</code> "). Default <code>method = "chaser"</code> .
<code>tuning</code>	a numeric value in general close to zero for the <code>rc</code> method and close to 1 for the <code>chaser</code> method. This argument control the step-length. Default <code>tuning = 1</code> .
<code>verbose</code>	a logical if <code>TRUE</code> print the values of the covariance parameters used on each iteration. Default <code>verbose = FALSE</code>
<code>weights</code>	Vector of weights for model fitting.

Value

A list with regression and covariance parameter estimates. Details about the estimation procedures as iterations, sensitivity, variability are also provided. In general the users do not need to use this function directly. The `mcglm` provides GLM interface for fitting `mcglm`.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. and Jorgensen, B. (2016) Multivariate covariance generalized linear models. *Journal of Royal Statistical Society - Series C* 65:649–675.

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`mcglm`, `mc_matrix_linear_predictor`, `mc_link_function` and `mc_variance_function`.

`gof`*Measures of Goodness-of-Fit*

Description

Extract the pseudo Gaussian log-likelihood (`plogLik`), pseudo Akaike Information Criterion (`pAIC`), pseudo Kullback-Leibler Information Criterion (`pKLIC`) and pseudo Bayesian Information Criterion (`pBIC`) for objects of `mcglm` class.

Usage

```
gof(object)
```

Arguments

`object` an object or a list of objects representing a model of `mcglm` class.

Value

Returns a data frame containing goodness-of-fit measures.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

Wang, M. (2014). Generalized Estimating Equations in Longitudinal Data Analysis: A Review and Recent Developments. *Advances in Statistics*, 1(1)1–13.

See Also

`plogLik`, `pAIC`, `pKLIC` and `pBIC`.

Description

Extract the Gosho Information Criterion (GOSHO) for an object of `mcglm` class. **WARNING:** This function is limited to models with ONE response variable. This function is general useful only for longitudinal data analysis.

Usage

```
GOSHO(object, id, verbose = TRUE)
```

Arguments

<code>object</code>	an object of <code>mcglm</code> class.
<code>id</code>	a vector which identifies the clusters or groups. The length and order of <code>id</code> should be the same as the number of observations. Data are assumed to be sorted so that observations on a cluster are contiguous rows for all entities in the formula.
<code>verbose</code>	logical. Print or not the GOSHO value.

Value

The value of the GOSHO criterion. Note that the function assumes that the data are in the correct order.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Wang, M. (2014). Generalized Estimating Equations in Longitudinal Data Analysis: A Review and Recent Developments. *Advances in Statistics*, 1(1)1–13.

See Also

`gof`, `plogLik`, `pAIC`, `pKLIC`, `ESS` and `RJC`.

Hunting

Hunting in Pico Basile, Bioko Island, Equatorial Guinea.

Description

Case study analysed in Bonat et. al. (2016) concernings on data of animals hunted in the village of Basile Fang, Bioko Norte Province, Bioko Island, Equatorial Guinea. Monthly number of blue duikers and other small animals shot or snared was collected for a random sample of 52 commercial hunters from August 2010 to September 2013. For each animal caught, the species, sex, method of capture and altitude were documented. The data set has 1216 observations.

- ALT - Factor five levels indicating the Altitude where the animal was caught.
- SEX - Factor two levels Female and Male.
- METHOD - Factor two levels Escopeta and Trampa.
- OT - Monthly number of other small animals hunted.
- BD - Monthly number of blue duikers hunted.
- OFFSET - Monthly number of hunter days.
- HUNTER - Hunter index.
- MONTH - Month index.
- MONTHCALENDAR - Month using calendar numbers (1-January, ..., 12-December).
- YEAR - Year calendar (2010–2013).
- HUNTER.MONTH - Index indicating observations taken at the same HUNTER and MONTH.

Usage

```
data(Hunting)
```

Format

a data.frame with 1216 records and 11 variables.

Source

Bonat, et. al. (2017). Modelling the covariance structure in marginal multivariate count models: Hunting in Bioko Island. *Journal of Agricultural Biological and Environmental Statistics*, 22(4):446–464.

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

Examples

```

library(mcglm)
library(Matrix)
data(Hunting, package="mcglm")
formu <- OT ~ METHOD*ALT + SEX + ALT*poly(MONTH, 4)
Z0 <- mc_id(Hunting)
Z1 <- mc_mixed(~0 + HUNTER.MONTH, data = Hunting)
fit <- mcglm(linear_pred = c(formu), matrix_pred = list(c(Z0, Z1)),
             link = c("log"), variance = c("poisson_tweedie"),
             power_fixed = c(FALSE),
             control_algorithm = list(max_iter = 100),
             offset = list(log(Hunting$OFFSET)), data = Hunting)
summary(fit)
anova(fit)

```

mcglm

*Fitting Multivariate Covariance Generalized Linear Models***Description**

The function `mcglm` is used to fit multivariate covariance generalized linear models. The models are specified by a set of lists giving a symbolic description of the linear and matrix linear predictors. The user can choose between a list of link, variance and covariance functions. The models are fitted using an estimating function approach, combining quasi-score functions for regression parameters and Pearson estimating function for covariance parameters. For details see Bonat and Jorgensen (2016).

Usage

```

mcglm(linear_pred, matrix_pred, link, variance, covariance,
      offset, Ntrial, power_fixed, data, control_initial,
      contrasts, weights, control_algorithm)

```

Arguments

<code>linear_pred</code>	a list of formula see formula for details.
<code>matrix_pred</code>	a list of known matrices to be used on the matrix linear predictor. For details see mc_matrix_linear_predictor .
<code>link</code>	a list of link functions names. Options are: "logit", "probit", "cauchit", "cloglog", "loglog", "identity", "log", "sqrt", "1/mu^2" and "inverse". See mc_link_function for details.
<code>variance</code>	a list of variance functions names. Options are: "constant", "tweedie", "poisson_tweedie", "binomialP" and "binomialPQ". See mc_variance_function for details.
<code>covariance</code>	a list of covariance link functions names. Options are: "identity", "inverse" and exponential-matrix "expm".

offset	a list of offset values if any.
Ntrial	a list of number of trials on Bernoulli experiments. It is useful only for binomialP and binomialPQ variance functions.
power_fixed	a list of logicals indicating if the values of the power parameter should be estimated or not.
data	a data frame.
control_initial	a list of initial values for the fitting algorithm. If no values are supplied automatic initial values will be provided by the function mc_initial_values .
contrasts	extra arguments to passed to model.matrix .
weights	A list of weights for model fitting. Each element of the list should be a vector of weights of size equals the number of observations. Missing observations should be annotated as NA.
control_algorithm	a list of arguments to be passed for the fitting algorithm. See fit_mcglm for details.

Value

mcglm returns an object of class 'mcglm'.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. and Jorgensen, B. (2016) Multivariate covariance generalized linear models. *Journal of Royal Statistical Society - Series C* 65:649–675.

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

[fit_mcglm](#), [mc_link_function](#) and [mc_variance_function](#).

mc_anova_disp

Anova Tables for dispersion components

Description

IT IS AN EXPERIMENTAL FUNCTION BE CAREFUL! Performs Wald tests of the significance for the dispersion components by response variables. This function is useful for joint hypothesis tests of dispersion coefficients associated with categorical covariates with more than two levels. It is not designed for model comparison.

Usage

```
mc_anova_disp(object, idx_list, names_list, ...)
```

Arguments

object	an object of class mcglm, usually, a result of a call to mcglm() function.
idx_list	list with indexes for parameter tests.
names_list	list of names to appear in the anova table.
...	additional arguments affecting the summary produced. Note that there is no extra options for mcglm object class.

Value

A data.frame with Chi-square statistic to test the null hypothesis of a parameter, or a set of parameters, be zero. Degree of freedom (Df) and p-values. The Wald test based on the observed covariance matrix of the parameters is used.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Examples

```
x1 <- seq(0, 1, l = 100)
x2 <- gl(5, 20)
beta <- c(5, 0, -2, -1, 1, 2)
X <- model.matrix(~ x1 + x2)
set.seed(123)
y <- rnorm(100, mean = 10, sd = X%%beta)
data = data.frame("y" = y, "x1" = x1, "x2" = x2, "id" = 1)
fit.anova <- mcglm(c(y ~ 1), list(mc_dglm(~ x1 + x2, id = "id", data)),
  control_algorithm = list(tuning = 0.9), data = data)
X <- model.matrix(~ x1 + x2, data = data)
idx <- attr(X, "assign")
idx_list <- list("idx" = idx, "idx" = idx)
names_list <- list(colnames(X), colnames(X))
mc_anova_disp(object = fit.anova, idx = idx_list, names_list = names_list)
```

mc_bias_corrected_std *Bias-corrected Standard Error for Regression Parameters*

Description

Compute bias-corrected standard error for regression parameters in the context of clustered observations for an object of mcglm class. It is also robust and has improved finite sample properties.

Usage

```
mc_bias_corrected_std(object, id)
```

Arguments

object	an object of mcglm class.
id	a vector which identifies the clusters. The length and order of id should be the same as the number of observations. The data set are assumed to be sorted so that observations on a cluster are contiguous rows for all entities.

Value

A variance-covariance matrix. Note that the function assumes that the data are in the correct order.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Nuamah, I. F. and Qu, Y. and Aminu, S. B. (1996). A SAS macro for stepwise correlated binary regression. *Computer Methods and Programs in Biomedicine* 49, 199–210.

See Also

mc_robust_std.

mc_car

Conditional Auto regressive Model Structure

Description

The function mc_car helps to build the components of the matrix linear predictor used for fitting conditional auto regressive models. This function is used in general for fitting spatial areal data using the well known conditional auto regressive models (CAR). This function depends on a list of neighbors, such a list can be constructed, for example using the tri2nb function from the spdep package based on spatial coordinates. This way to specify the matrix linear predictor can also be applied for spatial continuous data, as an approximation.

Usage

```
mc_car(list_neigh, intrinsic = FALSE)
```

Arguments

list_neigh	list of neighbors.
intrinsic	logical.

Value

A list of a matrix (`intrinsic = TRUE`) or two matrices (`intrinsic = FALSE`).

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`mc_id`, `mc_compute_rho`, `mc_conditional_test`, `mc_dist`, `mc_ma`, `mc_rw`
and `mc_mixed`.

mc_complete_data	<i>Complete data in case of missing observations</i>
------------------	--

Description

The function `mc_complete_data` completes a data set with NA values for helping to construct the components of the matrix linear predictor in models that require equal number of observations by subjects (`id`).

Usage

```
mc_complete_data(data, id, index, id.exp)
```

Arguments

<code>data</code>	a data.frame to be completed with NA.
<code>id</code>	name of the column (string) containing the subject id.
<code>index</code>	name of the column (string) containing the index to be completed.
<code>id.exp</code>	how the index is expected to be for all subjects.

Value

A data.frame with the same number of observations by subject.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. Journal of Statistical Software, 84(4):1–30.

See Also

mc_dglm, mc_ns, mc_ma and mc_rw.

mc_compute_rho

Autocorrelation Estimates

Description

Compute autocorrelation estimates based on a fitted model using the mc_car structure. The mcglm approach fits models using a linear covariance structure, but in general in this parametrization for spatial models the parameters have no simple interpretation in terms of spatial autocorrelation. The function mc_compute_rho computes the autocorrelation based on a fitted model.

Usage

```
mc_compute_rho(object, level = 0.975)
```

Arguments

object an object or a list of objects representing a model of mcglm class.
level the confidence level required.

Value

Returns estimate, standard error and confidential interval for the spatial autocorrelation parameter.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. Journal of Statistical Software, 84(4):1–30.

See Also

mc_car and mc_conditional_test.

mc_conditional_test *Conditional Hypotheses Tests*

Description

Compute conditional hypotheses tests for fitted `mcglm` model class. When fitting models with extra power parameters, the standard errors associated with the dispersion parameters can be large. In that cases, we suggest to conduct conditional hypotheses test instead of the orthodox marginal test for the dispersion parameters. The function `mc_conditional_test` offers an ease way to conduct such conditional test. Furthermore, the function is quite flexible and can be used for any other conditional hypotheses test.

Usage

```
mc_conditional_test(object, parameters, test, fixed)
```

Arguments

<code>object</code>	an object representing a model of <code>mcglm</code> class.
<code>parameters</code>	which parameters will be included in the conditional test.
<code>test</code>	index indicating which parameters will be tested given the values of the other parameters.
<code>fixed</code>	index indicating which parameters should be fixed on the conditional test.

Value

Returns estimates, conditional standard errors and Z-statistics.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

`mc_dglm`*Double Generalized Linear Models Structure*

Description

The function `mc_dglm` builds the components of the matrix linear predictor used for fitting double generalized linear models.

Usage

```
mc_dglm(formula, id, data)
```

Arguments

<code>formula</code>	a formula spefying the components of the covariance structure.
<code>id</code>	name of the column (string) containing the subject index. (If ts is not repeated measures, use <code>id = 1</code> for all observations).
<code>data</code>	data set.

Value

A list of a diagonal matrices, whose values are given by the covariates assumed to describe the covariance structure.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`mc_id`, `mc_dist`, `mc_ma`, `mc_rw`
and `mc_mixed`.

Description

The function `mc_dist` helps to build the components of the matrix linear predictor using matrices based on distances. This function is generally used for the analysis of longitudinal and spatial data. The idea is to use the inverse of some measure of distance as for example the Euclidean distance to model the covariance structure within response variables. The model can also use the inverse of distance squared or high order power.

Usage

```
mc_dist(id, time, data, method = "euclidean")
```

Arguments

<code>id</code>	name of the column (string) containing the subject index. For spatial data use the same <code>id</code> for all observations (one unit sample).
<code>time</code>	name of the column (string) containing the index indicating the time. For spatial data use the same index for all observations.
<code>data</code>	data set.
<code>method</code>	distance measure to be used.

Details

The distance measure must be one of "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". This function is a customize call of the `dist` function.

Value

A matrix of `dgCMatrix` class.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`dist`, `mc_id`, `mc_conditional_test`, `mc_car`, `mc_ma`, `mc_rw` and `mc_mixed`.

Examples

```
id <- rep(1:2, each = 4)
time <- rep(1:4, 2)
data <- data.frame("id" = id, "time" = time)
mc_dist(id = "id", time = "time", data = data)
mc_dist(id = "id", time = "time", data = data, method = "canberra")
```

mc_id

Independent Model Structure

Description

Builds an identity matrix to be used as a component of the matrix linear predictor. It is in general the first component of the matrix linear predictor, a kind of intercept matrix.

Usage

```
mc_id(data)
```

Arguments

data the data set to be used.

Value

A list of matrix.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. Journal of Statistical Software, 84(4):1–30.

See Also

mc_dist, mc_ma, mc_rw and mc_mixed.

 mc_initial_values *Automatic Initial Values*

Description

This function provides initial values to be used when fitting multivariate covariance generalized linear models by using the function `mcglm`. In general the users do not need to use this function, since it is already employed when setting the argument `control_initial = "automatic"` in the `mcglm` function. However, if the users want to change some of the initial values, this function can be useful.

Usage

```
mc_initial_values(linear_pred, matrix_pred, link, variance,
                  covariance, offset, Ntrial, contrasts, data)
```

Arguments

<code>linear_pred</code>	a list of formula see formula for details.
<code>matrix_pred</code>	a list of known matrices to be used on the matrix linear predictor. See mc_matrix_linear_predictor for details.
<code>link</code>	a list of link functions names, see mcglm for details.
<code>variance</code>	a list of variance functions names, see mcglm for details.
<code>covariance</code>	a list of covariance link functions names, see mcglm for details.
<code>offset</code>	a list of offset values if any.
<code>Ntrial</code>	a list of the number of trials on Bernoulli experiments. It is useful only for "binomialP" and "binomialPQ" variance functions.
<code>contrasts</code>	list of contrasts to be used in the model.matrix .
<code>data</code>	data frame.

Details

To obtain initial values for multivariate covariance generalized linear models the function `mc_initial_values` fits a generalized linear model (GLM) using the function `glm` with the specified linear predictor and link function for each response variables considering independent observations. The family argument is always specified as `quasi`. The link function depends on the specification of the argument `link`. The variance function is always specified as `"mu"` the only exception appears when using `variance = "constant"` then the family argument in the `glm` function is specified as `quasi(link = link, variance = "constant")`. The estimated value of the dispersion parameter from the `glm` function is used as initial value for the first component of the matrix linear predictor, for all other components the value zero is used.

For the cases `covariance = "inverse"` and `covariance = "expm"` the inverse and the logarithm of the estimated dispersion parameter is used as initial value for the first component of the matrix linear predictor. The value of the power parameter is always started at 1. In the cases of multivariate models the correlation between response variables is always started at 0.

Value

Return a list of initial values to be used while fitting in the `mcglm` function.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

mc_link_function *Link Functions*

Description

The `mc_link_function` is a customized call of the `make.link` function.

Given the name of a link function, it returns a list with two elements. The first element is the inverse of the link function applied on the linear predictor $\mu = g^{-1}(X\beta)$. The second element is the derivative of μ with respect to the regression parameters β . It will be useful when computing the quasi-score function.

Usage

```
mc_link_function(beta, X, offset, link)
```

```
mc_logit(beta, X, offset)
```

```
mc_probit(beta, X, offset)
```

```
mc_cauchit(beta, X, offset)
```

```
mc_cloglog(beta, X, offset)
```

```
mc_loglog(beta, X, offset)
```

```
mc_identity(beta, X, offset)
```

```
mc_log(beta, X, offset)
```

```
mc_sqrt(beta, X, offset)
```

```
mc_invmu2(beta, X, offset)
```

```
mc_inverse(beta, X, offset)
```

Arguments

`beta` a numeric vector of regression parameters.
`X` a design matrix, see `model.matrix` for details.

offset	a numeric vector of offset values. It will be sum up on the linear predictor as a covariate with known regression parameter equals one ($\mu = g^{-1}(X\beta + offset)$). If no offset is present in the model, set offset = NULL.
link	a string specifying the name of the link function. Options are: "logit", "probit", "cauchit", "cloglog", "loglog", "identity", "log", "sqrt", "1/mu^2" and inverse. A user defined link function can be used (see Details).

Details

The link function is an important component of the multivariate covariance generalized linear models, since it links the expectation of the response variable with the covariates. Let β be a (p x 1) regression parameter vector and X be an (n x p) design matrix. The expected value of the response variable Y is given by

$$E(Y) = g^{-1}(X\beta),$$

where g is the link function and $\eta = X\beta$ is the linear predictor. Let D be a (n x p) matrix whose entries are given by the derivatives of μ with respect to β . Such a matrix will be required for the fitting algorithm. The function `mc_link_function` returns a list where the first element is μ (n x 1) vector and the second is the D (n x p) matrix. A user defined function can also be used. It must be a function with arguments `beta`, `X` and `offset` (set to NULL if non needed). The function must return a length 2 named list with `mu` and `D` elements as a vector and a matrix of proper dimensions.

Value

A list with two elements: `mu` and `D` (see Details).

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

See Also

[model.matrix](#), [make.link](#).

Examples

```
x1 <- seq(-1, 1, l = 5)
X <- model.matrix(~ x1)
mc_link_function(beta = c(1,0.5), X = X,
                 offset = NULL, link = 'log')
mc_link_function(beta = c(1,0.5), X = X,
                 offset = rep(10,5), link = 'identity')
```

`mc_ma`*Moving Average Models Structure*

Description

The function `mc_ma` helps to build the components of the matrix linear predictor associated with moving average models. This function is generally used for the analysis of longitudinal and times series data. The user can specify the order of the moving average process.

Usage

```
mc_ma(id, time, data, order = 1)
```

Arguments

<code>id</code>	name of the column (string) containing the subject index. Note that this structure was designed to deal with longitudinal data. For times series data use the same <code>id</code> for all observations (one unit sample).
<code>time</code>	name of the column (string) containing the index indicating the time.
<code>data</code>	data set.
<code>order</code>	order of the moving average process.

Details

This function was designed mainly to deal with longitudinal data, but can also be used for times series analysis. In that case, the `id` argument should contain only one index. It pretends a longitudinal data taken just for one individual or unit sample. This function is a simple call of the [bandSparse](#) function from the `Matrix` package.

Value

A matrix of `dgCMatrix` class.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`mc_id`, `mc_dist`, `mc_car`, `mc_rw` and `mc_mixed`.

Examples

```
id <- rep(1:2, each = 4)
time <- rep(1:4, 2)
data <- data.frame("id" = id, "time" = time)
mc_ma(id = "id", time = "time", data = data, order = 1)
mc_ma(id = "id", time = "time", data = data, order = 2)
```

mc_matrix_linear_predictor

Matrix Linear Predictor

Description

Compute the matrix linear predictor. It is an internal function, however, since the concept of matrix linear predictor was proposed recently. I decided let this function visible to the interested reader gets some feeling about how it works.

Usage

```
mc_matrix_linear_predictor(tau, Z)
```

Arguments

tau	a numeric vector of dispersion parameters.
Z	a list of known matrices.

Details

Given a list with a set of known matrices (Z_0, \dots, Z_D) the function `mc_matrix_linear_predictor` returns $U = \tau_0 Z_0 + \dots + \tau_D Z_D$.

Value

A matrix.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

Bonat, W. H. and Jorgensen, B. (2016) Multivariate covariance generalized linear models. *Journal of Royal Statistical Society - Series C* 65:649–675.

See Also

mc_id, mc_dist, mc_ma, mc_rw, mc_mixed and mc_car.

Examples

```
require(Matrix)
Z0 <- Diagonal(5, 1)
Z1 <- Matrix(rep(1,5)%*%t(rep(1,5)))
Z <- list(Z0, Z1)
mc_matrix_linear_predictor(tau = c(1,0.8), Z = Z)
```

 mc_mixed

Mixed Models Structure

Description

The function `mc_mixed` helps to build the components of the matrix linear predictor associated with mixed models. It is useful to model the covariance structure as a function of known covariates in a linear mixed model fashion (Bonat, et. al. 2016). The `mc_mixed` function was designed to analyse repeated measures and longitudinal data, where in general the observations are taken at a fixed number of groups, subjects or unit samples.

Usage

```
mc_mixed(formula, data)
```

Arguments

formula	a formula model to build the matrix linear predictor. See details.
data	data set.

Details

The formula argument should be specified similar to the linear predictor for the mean structure, however the first component should be 0 and the second component should always indicate the name of the column containing the subject or unit sample index. It should be a factor. The other covariates are specified after a slash "/" in the usual way. For example, $\sim 0 + \text{SUBJECT}/(\text{x1} + \text{x2})$ means that the column SUBJECT contains the subject or unit sample index, while the covariates that can be continuous or factors are given in the columns x1 and x2. Be careful the parenthesis after the "/" are mandatory, when including more than one covariate. The special case where only the SUBJECT effect is requested the formula takes the form $\sim 0 + \text{SUBJECT}$ without any extra covariate. This structure corresponds to the well known compound symmetry structure. By default the function `mc_mixed` include all interaction terms, the users can ignore the interactions terms removing them from the matrix linear predictor.

Value

A list of matrices.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

Bonat, et. al. (2016). Modelling the covariance structure in marginal multivariate count models: Hunting in Bioko Island. *Journal of Agricultural Biological and Environmental Statistics*, 22(4):446–464.

See Also

mc_id, mc_conditional_test, mc_dist, mc_ma, mc_rw and mc_car.

Examples

```
SUBJECT <- gl(2, 6)
x1 <- rep(1:6, 2)
x2 <- rep(gl(2,3),2)
data <- data.frame(SUBJECT, x1 , x2)
# Compound symmetry structure
mc_mixed(~0 + SUBJECT, data = data)
# Compound symmetry + random slope for x1 and interaction or correlation
mc_mixed(~0 + SUBJECT/x1, data = data)
# Compound symmetry + random slope for x1 and x2 plus interactions
mc_mixed(~0 + SUBJECT/(x1 + x2), data = data)
```

mc_ns

Non-structure Model Structure

Description

The function `mc_non` builds the components of the matrix linear predictor used for fitting non-structured covariance matrix. In general this model is hard to fit due to the large number of parameters.

Usage

```
mc_ns(id, data, group = NULL, marca = NULL)
```

Arguments

id	name of the column (string) containing the subject index. Note this structure was designed to deal with longitudinal data. For times series or spatial data use the same id for all observations (one unit sample).
data	data set.
group	name of the column (string) containing a group specific for which the covariance should change.
marca	level (string) of the column group for which the covariance should change.

Value

A list of a $n*(n-1)/2$ matrices.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. Journal of Statistical Software, 84(4):1–30.

See Also

mc_id, mc_dglm, mc_dist, mc_ma, mc_rw
and mc_mixed.

mc_remove_na

Remove NA from Matrix Linear Predictor

Description

The function mc_remove_na removes NA from each component of the matrix linear predictor. It is in general used after the function mc_complete_data.

Usage

```
mc_remove_na(matrix_pred, cod)
```

Arguments

matrix_pred	a list of known matrices.
cod	index indicating the columns should be removed.

Value

A list of matrices.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

mc_dglm, mc_ns, mc_ma and mc_rw.

mc_robust_std

Robust Standard Error for Regression Parameters

Description

Compute robust standard error for regression parameters in the context of clustered observations for an object of mcglm class.

Usage

```
mc_robust_std(object, id)
```

Arguments

object	an object of mcglm class.
id	a vector which identifies the clusters or subject indexes. The length and order of id should be the same as the number of observations. Data are assumed to be sorted so that observations on a cluster are contiguous rows for all entities in the formula.

Value

A variance-covariance matrix. Note that the function assumes that the data are in the correct order.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Nuamah, I. F. and Qu, Y. and Aminu, S. B. (1996). A SAS macro for stepwise correlated binary regression. *Computer Methods and Programs in Biomedicine* 49, 199–210.

See Also

mc_bias_correct_std.

`mc_rw`*Random Walk Models Structure*

Description

The function `mc_rw` builds the components of the matrix linear predictor associated with random walk models. This function is generally used for the analysis of longitudinal and times series data. The user can specify the order of the random walk process.

Usage

```
mc_rw(id, time, data, order = 1, proper = FALSE)
```

Arguments

<code>id</code>	name of the column (string) containing the subject index. Note that this structure was designed to deal with longitudinal data. For times series data use the same <code>id</code> for all observations (one unit sample).
<code>time</code>	name of the column (string) containing the index indicating the time.
<code>data</code>	data set.
<code>order</code>	order of the random walk model.
<code>proper</code>	logical.

Value

If `proper = FALSE` a matrix of `dgCMatrix` class. If `proper = TRUE` a list with two matrices of `dgCMatrix` class.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`mc_id`, `mc_dist`, `mc_car`, `mc_ma`, `mc_mixed` and `mc_compute_rho`.

Examples

```
id <- rep(1:2, each = 4)
time <- rep(1:4, 2)
data <- data.frame("id" = id, "time" = time)
mc_rw(id = "id", time = "time", data = data, order = 1, proper = FALSE)
mc_rw(id = "id", time = "time", data = data, order = 1, proper = TRUE)
mc_rw(id = "id", time = "time", data = data, order = 2, proper = TRUE)
```

mc_sic

Score Information Criterion - Regression

Description

Compute the score information criterion (SIC) for an object of `mcglm` class. The SIC is useful for selecting the components of the linear predictor. It can be used to construct an stepwise covariate selection.

Usage

```
mc_sic(object, scope, data, response, penalty = 2, weights)
```

Arguments

<code>object</code>	an object of <code>mcglm</code> class.
<code>scope</code>	a vector of covariate names to be tested.
<code>data</code>	data set containing all variables involved in the model.
<code>response</code>	index indicating for which response variable the SIC should be computed.
<code>penalty</code>	penalty term (default = 2).
<code>weights</code>	Vector of weights for model fitting.

Value

A data frame containing SIC values, degree of freedom, Tu-statistics and chi-squared reference values.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

Bonat, et. al. (2016). Modelling the covariance structure in marginal multivariate count models: Hunting in Bioko Island. *Journal of Agricultural Biological and Environmental Statistics*, 22(4):446–464.

See Also

mc_sic_covariance.

Examples

```
set.seed(123)
x1 <- runif(100, -1, 1)
x2 <- gl(2,50)
beta = c(5, 0, 3)
X <- model.matrix(~ x1 + x2)
y <- rnorm(100, mean = X%%beta , sd = 1)
data <- data.frame(y, x1, x2)
# Reference model
fit0 <- mcglm(c(y ~ 1), list(mc_id(data)), data = data)
# Computing SIC
mc_sic(fit0, scope = c("x1","x2"), data = data, response = 1)
```

mc_sic_covariance *Score Information Criterion - Covariance*

Description

Compute the score information criterion (SIC) for an object of mcglm class. The SIC-covariance is useful for selecting the components of the matrix linear predictor. It can be used to construct an stepwise procedure to select the components of the matrix linear predictor.

Usage

```
mc_sic_covariance(object, scope, idx, data, penalty = 2, response, weights)
```

Arguments

object	an object of mcglm class.
scope	a list of matrices to be tested.
idx	indicator of matrices belong to the same effect. It is useful for the case where more than one matrix represents the same effect.
data	data set containing all variables involved in the model.
penalty	penalty term (default = 2).
response	index indicating for which response variable SIC-covariance should be computed.
weights	Vector of weights for model fitting.

Value

A data frame containing SIC-covariance values, degree of freedom, Tu-statistics and chi-squared reference values for each matrix in the scope argument.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, et. al. (2016). Modelling the covariance structure in marginal multivariate count models: Hunting in Bioko Island. *Journal of Agricultural Biological and Environmental Statistics*, 22(4):446–464.

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

mc_sic.

Examples

```
set.seed(123)
SUBJECT <- gl(10, 10)
y <- rnorm(100)
data <- data.frame(y, SUBJECT)
Z0 <- mc_id(data)
Z1 <- mc_mixed(~0+SUBJECT, data = data)
# Reference model
fit0 <- mcglm(c(y ~ 1), list(Z0), data = data)
# Testing the effect of the matrix Z1
mc_sic_covariance(fit0, scope = Z1, idx = 1,
  data = data, response = 1)
# As expected Tu < Chisq indicating non-significance of Z1 matrix
```

mc_twin

Twin Models Structure

Description

The function `mc_twin` helps to build the components of the matrix linear predictor associated with ACDE models for analysis of twin data.

Usage

```
mc_twin(id, twin.id, type, replicate = NULL, structure, data)
```

```
mc_twin_bio(id, twin.id, type, replicate = NULL, structure, data)
```

```
mc_twin_full(id, twin.id, type, replicate, formula, data)
```

Arguments

id	name of the column (string) containing the twin index. It should be the same index (number) for both twins.
twin.id	name of the column (string) containing the twin index inside the pair. In general 1 for the first twin and 2 for the second twin.
type	name of the column (string) containing the indication of the twin as mz or dz. It should be a factor with only two levels mz and dz. Be sure that the reference level is mz.
replicate	name of the column (string) containing the index for more than one observation taken at the same twin pair. It is used for example in twin longitudinal studies. In that case, the replication column should contain the time index.
structure	model type options are full, flex, uns, ACE, ADE, AE, CE and E. See example for details.
data	data set.
formula	internal.

Value

A list of matrices of `dgMatrix` class.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`mc_id`, `mc_dist`, `mc_car`, `mc_rw`, `mc_ns`, `mc_dglm` and `mc_mixed`.

Examples

```
id <- rep(1:5, each = 4)
id.twin <- rep(1:2, 10)
```

 mc_variance_function *Variance Functions*

Description

Compute the variance function and its derivatives with respect to regression, dispersion and power parameters.

Usage

```
mc_variance_function(mu, power, Ntrial, variance, inverse,
                    derivative_power, derivative_mu)
```

```
mc_power(mu, power, inverse, derivative_power, derivative_mu)
```

```
mc_binomialP(mu, power, inverse, Ntrial,
             derivative_power, derivative_mu)
```

```
mc_binomialPQ(mu, power, inverse, Ntrial,
              derivative_power, derivative_mu)
```

Arguments

mu	a numeric vector. In general the output from mc_link_function .
power	a numeric value (power and binomialP) or a vector (binomialPQ) of the power parameters.
Ntrial	number of trials, useful only when dealing with binomial response variables.
variance	a string specifying the name (power, binomialP or binomialPQ) of the variance function.
inverse	logical. Compute the inverse or not.
derivative_power	logical if compute (TRUE) or not (FALSE) the derivatives with respect to the power parameter.
derivative_mu	logical if compute (TRUE) or not (FALSE) the derivative with respect to the mu parameter.

Details

The function `mc_variance_function` computes three features related with the variance function. Depending on the logical arguments, the function returns $V^{1/2}$ and its derivatives with respect to the parameters power and mu, respectively. The output is a named list, completely informative about what the function has been computed. For example, if `inverse = FALSE`, `derivative_power = TRUE` and `derivative_mu = TRUE`. The output will be a list, with three elements: `V_sqrt`, `D_V_sqrt_power` and `D_V_sqrt_mu`.

Value

A list with from one to four elements depends on the arguments.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. and Jorgensen, B. (2016) Multivariate covariance generalized linear models. *Journal of Royal Statistical Society - Series C* 65:649–675.

See Also

[mc_link_function](#).

Examples

```
x1 <- seq(-1, 1, l = 5)
X <- model.matrix(~x1)
mu <- mc_link_function(beta = c(1, 0.5), X = X, offset = NULL,
                      link = "logit")
mc_variance_function(mu = mu$mu, power = c(2, 1), Ntrial = 1,
                   variance = "binomialPQ", inverse = FALSE,
                   derivative_power = TRUE, derivative_mu = TRUE)
```

NewBorn

Respiratory Physiotherapy on Premature Newborns.

Description

The NewBorn dataset consists of a prospective study to assess the effect of respiratory physiotherapy on the cardiopulmonary function of ventilated preterm newborn infants with birth weight lower than 1500 g. The data set was collected and kindly made available by the nursing team of the Waldemar Monastier hospital, Campo Largo, PR, Brazil. The NewBorn dataset was analysed in Bonat and Jorgensen (2016) as an example of mixed outcomes regression model.

- Sex - Factor two levels Female and Male.
- GA - Gestational age (weeks).
- BW - Birth weight (mm).
- APGAR1M - APGAR index in the first minute of life.
- APGAR5M - APGAR index in the fifth minute of life.
- PRE - Factor, two levels (Premature: YES; NO).
- HD - Factor, two levels (Hansen's disease, YES; NO).
- SUR - Factor, two levels (Surfactant, YES; NO).

- JAU - Factor, two levels (Jaundice, YES; NO).
- PNE - Factor, two levels (Pneumonia, YES; NO).
- PDA - Factor, two levels (Persistence of ductus arteriosus, YES; NO).
- PPI - Factor, two levels (Primary pulmonary infection, YES; NO).
- OTHERS - Factor, two levels (Other diseases, YES; NO).
- DAYS - Age (days).
- AUX - Factor, two levels (Type of respiratory auxiliary, HOOD; OTHERS).
- RR - Respiratory rate (continuous).
- HR - Heart rate (continuous).
- SPO2 - Oxygen saturation (bounded).
- TREAT - Factor, three levels (Respiratory physiotherapy, Evaluation 1; Evaluation 2; Evaluation 3).
- NBI - Newborn index.
- TIME - Days of treatment.

Usage

```
data(NewBorn)
```

Format

a data.frame with 270 records and 21 variables.

Source

Bonat, W. H. and Jorgensen, B. (2016) Multivariate covariance generalized linear models. *Journal of Royal Statistical Society - Series C* 65:649–675.

Examples

```
library(mcglm)
library(Matrix)
data(NewBorn, package="mcglm")
formu <- SPO2 ~ Sex + APGAR1M + APGAR5M + PRE + HD + SUR
Z0 <- mc_id(NewBorn)
fit <- mcglm(linear_pred = c(formu), matrix_pred = list(Z0),
            link = c("logit"), variance = c("binomialP"),
            power_fixed = c(TRUE),
            data = NewBorn,
            control_algorithm = list(verbose = FALSE, tuning = 0.5))
summary(fit)
```

pAIC

Pseudo Akaike Information Criterion

Description

Extract the pseudo Akaike information criterion (pAIC) for objects of `mcglm` class.

Usage

```
pAIC(object, verbose = TRUE)
```

Arguments

`object` an object or a list of objects representing a model of `mcglm` class.
`verbose` logical. Print or not the pAIC value.

Value

Returns the value of the pseudo Akaike information criterion (pAIC).

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

See Also

`gof`, `plogLik`, `ESS`, `pKLIC`, `GOSH0` and `RJC`.

pBIC

Pseudo Bayesian Information Criterion

Description

Extract the pseudo Bayesian information criterion (pBIC) for objects of `mcglm` class.

Usage

```
pBIC(object, verbose = TRUE)
```


Arguments

object an object or a list of objects representing a model of mcglm class.
verbose logical. Print or not the pBIC value.

Value

Returns the value of the pseudo Bayesian information criterion (pBIC).

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. Journal of Statistical Software, 84(4):1–30.

See Also

gof, plogLik, ESS, pKLIC, GOSH0 and RJC.

pKLIC

Pseudo Kullback-Leibler Information Criterion

Description

Extract the pseudo Kullback-Leibler information criterion (pKLIC) for objects of mcglm class.

Usage

```
pKLIC(object, verbose = TRUE)
```

Arguments

object an object or a list of objects representing a model of mcglm class.
verbose logical. Print or not the pKLIC value.

Value

Returns the value of the pseudo Kullback-Leibler information criterion.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. Journal of Statistical Software, 84(4):1–30.

See Also

gof, plogLik, ESS, pAIC, GOSHO and RJC.

plogLik *Gaussian Pseudo-loglikelihood*

Description

Extract the Gaussian pseudo-loglikelihood (plogLik) value for objects of mcglm class.

Usage

```
plogLik(object, verbose = TRUE)
```

Arguments

object an object or a list of objects representing a model of mcglm class.
verbose logical. Print or not the plogLik value.

Value

Returns the value of the Gaussian pseudo-loglikelihood.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

plot.mcglm *Residuals and algorithm check plots*

Description

Residual and algorithm check analysis for objects of mcglm class.

Usage

```
## S3 method for class 'mcglm'  
plot(x, type = "residuals", ...)
```

Arguments

x a fitted mcglm object.

type specify which graphical analysis will be performed. Options are: "residuals" and "algorithm".

... additional arguments affecting the plot produced. Note that there is no extra options for mcglm object class.

Value

The function plot.mcglm was designed to offer a fast residuals analysis based on the Pearson residuals. Current version offers a simple Pearson residuals versus fitted values and a quantile plot. When using algorithm = TRUE the function will plot a summary of the fitting algorithm shows the trajectory or iterations of the fitting algorithm. The iterations are shown in terms of values of the model parameters and also the actually value of the quasi-score and Pearson estimating functions. Hence, a quickly check of the algorithm convergence is obtained.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

See Also

residuals and fitted.

print.mcglm

Print

Description

The default print method for an object of mcglm class.

Usage

```
## S3 method for class 'mcglm'
print(x, ...)
```

Arguments

x fitted model objects of class mcglm as produced by mcglm().

... further arguments passed to or from other methods.

Value

No return value.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

See Also

summary.

residuals.mcglm	<i>Residuals</i>
-----------------	------------------

Description

Compute residuals for an object of mcglm class.

Usage

```
## S3 method for class 'mcglm'  
residuals(object, type = "raw", ...)
```

Arguments

object	an object of mcglm class.
type	the type of residuals which should be returned. Options are: "raw" (default), "pearson" and "standardized".
...	additional arguments affecting the residuals produced. Note that there is no extra options for mcglm object class.

Value

The function residuals.mcglm returns a matrix of residuals values.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

See Also

fitted.

RJC

Rotnitzky-Jewell Information Criterion

Description

Compute the Rotnitzky-Jewell information criterion for an object of `mcglm` class. **WARNINGS:** This function is limited to models with ONE response variable.

Usage

```
RJC(object, id, verbose = TRUE)
```

Arguments

<code>object</code>	an object of <code>mcglm</code> class.
<code>id</code>	a vector which identifies the clusters. The length and order of <code>id</code> should be the same as the number of observations. Data are assumed to be sorted so that observations on a cluster are contiguous rows for all entities in the formula.
<code>verbose</code>	logical. Print or not the RJC value.

Value

The value of the Rotnitzky-Jewell information criterion. Note that the function assumes that the data are in the correct order.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Source

Wang, M. (2014). Generalized Estimating Equations in Longitudinal Data Analysis: A Review and Recent Developments. *Advances in Statistics*, 1(1)1–13.

See Also

`gof`, `plogLik`, `pAIC`, `pKLIC`, `ESS` and `GOSH0`.

soil

Soil Chemistry Properties Data

Description

Soil chemistry properties measured on a regular grid with 10 x 25 points spaced by 5 meters.

- COORD.X - X coordinate.
- COORD.Y - Y coordinate.
- SAND - Sand portion of the sample.
- SILT - Silt portion of the sample.
- CLAY - Clay portion of the sample.
- PHWATER - Soil pH at water.
- CA - Calcium content.
- MG - Magnesium content.
- K - Potassio content.

Usage

```
data(soil)
```

Format

a data.frame with 250 records and 9 variables.

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The mcglm Package. *Journal of Statistical Software*, 84(4):1–30.

Examples

```
library(mcglm)
library(Matrix)
data(soil, package="mcglm")
Z1 <- mc_id(soil)
# Linear predictor
form.ca <- CA ~ COORD.X*COORD.Y + SAND + SILT + CLAY + PHWATER
fit.ca <- mcglm(linear_pred = c(form.ca), matrix_pred = list(Z1),
  link = "log", variance = "tweedie", covariance = "inverse",
  power_fixed = TRUE, data = soil,
  control_algorithm = list(max_iter = 1000, tuning = 0.1,
  verbose = FALSE, tol = 1e-03))
```

soya

Soybeans

Description

Experiment carried out in a vegetation house with soybeans. The experiment has two plants by plot with three levels of the factor amount of water in the soil (*water*) and five levels of potassium fertilization (*pot*). The plots were arranged in five blocks (*block*). Three response variables are of the interest, namely, grain yield, number of seeds and number of viable peas per plant. The data set has 75 observations of 7 variables.

- *pot* - Factor five levels of potassium fertilization.
- *water* - Factor three levels of amount of water in the soil.
- *block* - Factor five levels.
- *grain* - Continuous - Grain yield per plant.
- *seeds* - Count - Number of seeds per plant.
- *viablepeas* - Binomial - Number of viable peas per plant.
- *totalpeas* - Binomial - Total number of peas per plant.

Usage

```
data(soya)
```

Format

a `data.frame` with 75 records and 7 variables.

Source

Bonat, W. H. (2018). Multiple Response Variables Regression Models in R: The `mcglm` Package. *Journal of Statistical Software*, 84(4):1–30.

Examples

```
library(mcglm)
library(Matrix)
data(soya, package="mcglm")
formu <- grain ~ block + factor(water) * factor(pot)
Z0 <- mc_id(soya)
fit <- mcglm(linear_pred = c(formu), matrix_pred = list(Z0),
             data = soya)
anova(fit)
```

`summary.mcglm`*Summarizing*

Description

The default summary method for an object of mcglm class.

Usage

```
## S3 method for class 'mcglm'
summary(
  object,
  verbose = TRUE,
  print = c("Regression", "power", "Dispersion", "Correlation"),
  ...
)
```

Arguments

<code>object</code>	an object of mcglm class.
<code>verbose</code>	logical. Print or not the model summary.
<code>print</code>	print only part of the model summary, options are Regression, power, Dispersion and Correlation.
<code>...</code>	additional arguments affecting the summary produced. Note the there is no extra options for mcglm object class.

Value

Print a mcglm object.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

See Also

`print.`

`vcov.mcglm`*Variance-Covariance Matrix*

Description

Returns the variance-covariance matrix for an object of `mcglm` class.

Usage

```
## S3 method for class 'mcglm'  
vcov(object, ...)
```

Arguments

<code>object</code>	an object of <code>mcglm</code> class.
<code>...</code>	additional arguments affecting the summary produced. Note that there is no extra options for <code>mcglm</code> object class.

Value

A variance-covariance matrix.

Author(s)

Wagner Hugo Bonat, <wbonat@ufpr.br>

Index

- * **datasets**
 - ahs, 3
 - Hunting, 12
 - NewBorn, 38
 - soil, 46
 - soya, 47
- ahs, 3
- anova.mcglm, 4
- bandSparse, 26
- coef.mcglm, 5
- confint.mcglm, 6
- dist, 21
- ESS, 7
- fit_mcglm, 8, 14
- fitted.mcglm, 7
- formula, 13, 23
- gof, 10
- GOSHO, 11
- Hunting, 12
- make.link, 24, 25
- mc_anova_disp, 14
- mc_bias_corrected_std, 15
- mc_binomialP (mc_variance_function), 37
- mc_binomialPQ (mc_variance_function), 37
- mc_car, 16
- mc_cauchit (mc_link_function), 24
- mc_cloglog (mc_link_function), 24
- mc_complete_data, 17
- mc_compute_rho, 18
- mc_conditional_test, 19
- mc_dglm, 20
- mc_dist, 21
- mc_id, 22
- mc_identity (mc_link_function), 24
- mc_initial_values, 14, 23
- mc_inverse (mc_link_function), 24
- mc_invmu2 (mc_link_function), 24
- mc_link_function, 8, 13, 24, 37, 38
- mc_log (mc_link_function), 24
- mc_logit (mc_link_function), 24
- mc_loglog (mc_link_function), 24
- mc_ma, 26
- mc_matrix_linear_predictor, 13, 23, 27
- mc_mixed, 28
- mc_ns, 29
- mc_power (mc_variance_function), 37
- mc_probit (mc_link_function), 24
- mc_remove_na, 30
- mc_robust_std, 31
- mc_rw, 32
- mc_sic, 33
- mc_sic_covariance, 34
- mc_sqrt (mc_link_function), 24
- mc_twin, 35
- mc_twin_bio (mc_twin), 35
- mc_twin_full (mc_twin), 35
- mc_variance_function, 8, 13, 37
- mcglm, 9, 13, 23
- model.matrix, 8, 14, 23–25
- NewBorn, 38
- pAIC, 40
- pBIC, 40
- pKLIC, 41
- plogLik, 42
- plot.mcglm, 42
- print.mcglm, 43
- residuals.mcglm, 44
- RJC, 45
- soil, 46

soya, [47](#)

summary.mcglm, [48](#)

vcov.mcglm, [49](#)