Package 'GWRM'

July 31, 2017

Type Package

Title Generalized Waring Regression Model for Count Data

Version 2.1.0.3

Date 2017-07-18

Maintainer Antonio Jose Saez-Castillo <ajsaez@ujaen.es>

Depends R (>= 3.0.0)

Imports foreach, doParallel, stats, parallel

Description Statistical functions to fit, validate and describe a Generalized Waring Regression Model (GWRM).

License GPL-2 | GPL-3

LazyData true

NeedsCompilation no

Author Antonio Jose Saez-Castillo [aut, cre],
Silverio Vilchez-Lopez [aut],
Maria Jose Olmo-Jimenez [aut],
Jose Rodriguez-Avi [aut],
Antonio Conde-Sanchez [aut],
Ana Maria Martinez-Rodriguez [aut]

Repository CRAN

Date/Publication 2017-07-31 10:03:57 UTC

R topics documented:

---

add1.gw *Add All Possible Single Terms to a GWRM Model*

Description

Compute all the single terms in the scope argument that can be added to the GWRM model, fit those models and compute a table of the changes in fit.

Usage

```
## S3 method for class 'gw'
add1(object, scope, test = c("none", "Chisq"), k = 2,
  trace = FALSE, ...)
```

Arguments

object	a fitted object of class inheriting from "gw".
scope	a formula giving the terms to be considered for adding.
test	"none", which considers the AIC criterion, or Chisq, which is the likelihood-ratio test.
k	the penalty constant in AIC / Cp.
trace	if TRUE, print out progress reports.
...	further arguments passed to or from other methods.

Value

An object of class "anova" summarizing the differences in fit between the models.

Examples

```
data(goals)
fit0 <- gw(goals ~ offset(log(played)), data = goals)
summary(fit0)

fit1 <- add1(fit0, ~ position)
summary(fit1)
```

drop1.gw *Drop All Possible Single Terms to a GWRM Model*

Description

Compute all the single terms in the scope argument that can be dropped from the GWRM model, fit those models and compute a table of the changes in fit.

Usage

```
## S3 method for class 'gw'
drop1(object, scope, test = c("none", "Chisq"), k = 2,
  trace = FALSE, ...)
```

Arguments

object	a fitted object of class inheriting from "gw".
scope	a formula giving the terms to be considered for dropping.
test	"none", which considers the AIC criterion, or Chisq, which is the likelihood-ratio test.
k	the penalty constant in AIC / Cp.
trace	if TRUE, print out progress reports.
...	further arguments passed to or from other methods.

Value

An object of class "anova" summarizing the differences in fit between the models.

Examples

```
data(goals)

fit0 <- gw(goals ~ offset(log(played)), data = goals)
summary(fit0)

fit1 <- step(fit0, ~ position)
summary(fit1)
```

goals *Goals scored by footballers in the first division of the Spanish league*

Description

The response variable goals, is the number of goals scored by the footballers (excluding goalkeepers) in the first division of the Spanish league from the 2000/2001 to the 2006/2007 seasons. Since there are footballers who played more than one season, the season in which each one has played more matches has been selected. The covariates considered are the final classification of the team in each season, the position in the field (forward, midfielder and defender) and the number of matches played.

Usage

```
data(goals)
```

Format

A data frame with 1224 observations on the following 4 variables

Details

- clasif a numeric vector
- position a factor with levels Defender Forward Midfielder
- played a numeric vector
- goals a numeric vector

Source

MARCA sports paper

References

Rodriguez-Avi, J., Conde-Sanchez, A., Saez-Castillo, A. J., Olmo-Jimenez, M. J. and Martinez Rodriguez, A. M.(2009). A generalized Waring regression model for count data. Computational Statistics and Data Analysis, 53, pp. 3717-3725.

gw *Fitting GWRM Models*

Description

gw is used to fit Generalized Waring Regression Models (GWRM), specified by giving a symbolic description of the linear predictor.

Usage

```
gw(formula, data, weights, k = NULL, subset, na.action, kstart = 1,
  rostart = 2, betastart = NULL, offset, control = list(...),
  method = NULL, hessian = TRUE, model = TRUE, x = FALSE, y = TRUE,
  ...)
```

Arguments

formula	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula).
weights	an optional vector of 'prior weights' to be used in the fitting process. Should be NULL or a numeric vector.
k	optional value for the k parameter. If NULL, it is estimated.
subset	an optional vector specifying a subset of observations to be used in the fitting process.
na.action	a function which indicates what should happen when the data contain NA values. See glm.
kstart	starting value for the k parameter.
rostart	starting value for the ro parameter.
betastart	starting values for the vector of means.
offset	this can be used to specify an a priori known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases. One or more offset terms can be included in the formula instead or as well, and if more than one is specified their sum is used. See model.offset.
control	a list of parameters for controlling the fitting process.
method	the method to be used in fitting the model. The default method initially uses non-linear minimization (nlm) and Nelder-Mead optimization (optim) to fit a model which is then re-fitted by "L-BFGS-B" (optim). In this way, SE estimates for all the model parameters are provided. "nlm" and "Nelder-Mead" are also possible values, but they do not provide SE estimates for k and ro.

hessian	if TRUE, the hessian of f at the minimum is returned.
model	a logical value indicating whether model frame should be included as a component of the returned value.
x,y	For gw: logical values indicating whether the response vector and model matrix used in the fitting process should be returned as components of the returned value. For gw.fit: x is a design matrix of dimension n * p, and y is a vector of observations of length n.
...	further arguments.

Value

gw returns an object of class "gw". The function summary can be used to obtain or print a summary of the results. An object of class "gw" is a list containing the following components:

- Y if requested (the default), the y vector used.
- W the weights supplied, a vector of 1s if none were.
- covars names of the covariates in the model.
- nobs number of observations.
- covoffset a logical value specifying if an offset is present.
- loglik the maximized log-likelihood.
- aic a version of Akaike's *An Information Criterion*, minus twice the maximized log-likelihood plus twice the number of parameters.
- bic Bayesian Information Criterion, minus twice the maximized log-likelihood plus the number of parameters multiplied by the logarithm of the number of observations.
- df.residual the residual degrees of freedom.
- residuals the residuals in the final iteration of the fit.
- coefficients a named vector of coefficients.
- betaIIpars parameters estimates of the BetaII distribution.
- betascoefs a vector of coefficients.
- fitted.values the fitted mean values, obtained by transforming the linear predictors by the inverse of the link function.
- hessian a symmetric matrix giving an estimate of the Hessian at the solution found in the optimization of the log-likelihood function.
- cov an estimate of the covariance matrix of the model coefficients.
- se a vector of the standard errors estimates of the estimated coefficients.
- corr an estimate of the correlation matrix of the model coefficients.
- code a code that indicates successful convergence of the fitter function used (see nlm and optim helps).
- converged logical value that indicates if the optimization algorithms succesfull.
- method the name of the fitter function used.

- k if requested, the k value used.
- kBool a logical value specifying whether there is a k value or it is estimated.
- call the matched call.
- formula the formula supplied.
- terms the terms object used.
- data the data argument.
- offset the offset vector used.
- control the value of the control argument used.
- method the name of the fitter function used.
- contrasts (where relevant) the contrasts used.
- xlevels (where relevant) a record of the levels of the factors used in fitting.

Examples

```
data(goals)
gw(goals ~ position + offset(log(played)), data = goals)
```

GWRM *Generalized Waring Regression Model for Count Data*

Description

Statistical functions to fit, validate and describe a Generalized Waring Regression Model (GWRM).

Details

GWRM is a package for fitting and computing Generalized Waring Regression Models. It includes functions for fitting the model to data, for diagnosis and for computing important statistics, as those related to the partition of the variance. It also includes the dataset and the example of Rodriguez-Avi et al. (2009). The main function you're likely to need from **GWRM** is gw, in order to obtain a GWRM fit from data.

References

Rodriguez-Avi, J., Conde-Sanchez, A., Saez-Castillo, A. J., Olmo-Jimenez, M. J. and Martinez Rodriguez, A. M.(2009). A generalized Waring regression model for count data. *Computational Statistics and Data Analysis*, **53**, pp. 3717-3725.

partvar *Partition of the variance in a GWRM model*

Description

In a GWRM model, the variance may be split into three terms. The first component of this decomposition represents the variability due to randomness and it comes from the underlying Poisson model. The other two components refer to the variability that is not due to randomness but is explained by the presence of liability and proneness, respectively.

Usage

```
partvar(object, newdata = NULL, ...)
```

Arguments

object	an object class "gw" for which the partition is desired.
newdata	optionally, a data frame in which to look for variables with which to obtain the partition. If omitted, all the cases are used.
...	further arguments passed to or from other methods.

Details

One of the main drawbacks of using the Univariate Generalized Waring Distribution with parameters a, k and ro is that the parameters a and k are interchangeable when there is no auxiliary information given by the covariates. This identification problem prevents liability and proneness components from being distinguished in the univariate fits. To solve it, Irwin (1968) proposed that the expert should deduce which of these components is which from their own knowledge of the phenomenon. Xekalaki (1984) proposed a less subjective solution, developing a bivariate model that divides the observation period into two non-overlapping subperiods in which the model for proneness does not change. In the GWRM with, at least, one covariate, the parameters a and k are not interchangeable because, as in Xekalaki's bivariate model, the random model for proneness does not change. So, the identification problem of the non-random components is solved.

Value

Two data frames, with ratio of sources of variation and sources of variation in which variance is splitted.

Examples

```
data(goals)
fit <- gw(goals ~ position, data = goals)
pos <- factor(c("Defender", "Midfielder"), levels = c("Defender", "Midfielder", "Forward"))
lev <- data.frame(position = pos, played = c(17, 21))

partvar(fit, newdata = lev)
```

predict.gw *GWRM Predictions*

Description

Obtains predictions from a fitted GWRM object.

Usage

```
## S3 method for class 'gw'
predict(object = NULL, newdata = NULL, ...)
```

Arguments

object	a fitted object of class inheriting from "gw".
newdata	optionally, a data frame in which to look for variables with which to predict. If omitted, the fitted linear predictors are used.
...	further arguments passed to or from other methods.

Value

A data frame with newdata and their fitted means.

Examples

```
data(goals)
fit <- gw(goals ~ position, data = goals)
predict(fit)
```

---

residuals.gw *Extract and Visualize GWRM Model Residuals*

Description

residuals is a method which extracts model residuals from "gw", commonly returned by gw function. Optionally, it produces a normal plot with a simulated envelope of the residuals.

Usage

```
## S3 method for class 'gw'
residuals(object, type = "pearson", rep = 19,
  envelope = FALSE, title = "Simulated Envelope of Residuals",
  trace = FALSE, parallel = TRUE, ncores = 2, ...)
```

Arguments

object	object of class "gw" holding the fitted model
type	type of residuals to be extracted. Default is "pearson". "response" and "deviance" are also available. Deviance residuals are defined as $2[lnf(y_i\|y_i) - lnf(\widehat{\mu}_i\|y_i)]$, so that their sum is the value of the deviance statistic.
rep	number of replications for envelope construction. Default is 19, that is the smallest 95 percent band that can be built.
envelope	a logical value to specify if the envelope is required.
title	a title for the envelope.
trace	if TRUE a sort of information is printed during the running time.
parallel	if TRUE use parallel executation.
ncores	is the number of cores that we use if parallel is TRUE.
...	further arguments passed to or from other methods.

Details

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

Value

Residuals values and plot

Examples

```
data(goals)
set.seed(1)
fit0 <- gw(goals ~ position, data = goals[sample(1:nrow(goals), 75), ])
residuals(fit0, type = "pearson", rep = 19, envelope = TRUE, trace = FALSE, ncores = 2)
```

---

rgw *Simulation of Generalized Waring values*

Description

Random generation of values from a Generalized Waring distribution with parameters a, k and ro.

Usage

```
rgw(n, a, k, ro)
```

Arguments

n	number of random values to return.
a	vector of (non-negative) first parameters.
k	vector of (non-negative) second parameters.
ro	vector of (non-negative) third parameters.

Value

rgw is an auxiliar function which generates random samples from a Generalized Waring distribution to be used in the simulated envelope called by residuals.

Index