Package ‘gnlm’
February 3, 2019

Version 1.1.1
Title Generalized Nonlinear Regression Models
Depends R (>= 1.4), rmutil
Description A variety of functions to fit linear and nonlinear regression with a large selection of distributions.
License GPL (>= 2)
URL http://www.commanster.eu/rcode.html
BugReports https://github.com/swihart/gnlm/issues
Encoding UTF-8
LazyData true
LazyLoad true
RoxygenNote 6.1.1
NeedsCompilation yes
Author Bruce Swihart [cre, aut],
Jim Lindsey [aut] (Jim created this package, Bruce is maintaining the CRAN version)
Maintainer Bruce Swihart <bruce.swihart@gmail.com>
Repository CRAN
Date/Publication 2019-02-03 05:13:15 UTC

R topics documented:

bnlr ................................................................. 2
fit.dist .......................................................... 4
fmr ................................................................. 5
gnlr ............................................................... 9
gnlr3 ............................................................. 13
nlr ................................................................. 16
nordr ............................................................. 18
ordglm ........................................................... 21
rs2 ................................................................. 22
rs3 ................................................................. 23
bnlr fits user-specified nonlinear regression equations to binomial data with various link functions (logit, probit, comp log log, log log, Cauchy, Student t, stable, or mixture). The mixture link is a logistic link with extra probability mass for y=0 and y=n.

Usage

bnlr(y = NULL, link = "logit", mu = NULL, linear = NULL, pmu = NULL, pshape = NULL, wt = 1, envir = parent.frame(), print.level = 0, typsize = abs(p), ndigit = 10, gradtol = 1e-05, stepmax = 10 * sqrt(p *%*% p), steptol = 1e-05, iterlim = 100, fscale = 1)

Arguments

y
A two column matrix of binomial data or censored data or an object of class, response (created by restovec) or repeated (created by rmna or lvna). If the repeated data object contains more than one response variable, give that object in envir and give the name of the response variable to be used here.

link
A character string containing the name of the link function. The Student t, stable, and mixture links contain an unknown parameter to be estimated, respectively the logarithm of the degrees of freedom, the tail parameter transformed by log(tail/(2-tail)), and logit of the mixture probability, so that they lie on the whole real line.

mu
A user-specified function of pmu, and possibly linear, giving the regression equation for the location. This may contain a linear part as the second argument to the function. It may also be a formula beginning with ~, specifying either a linear regression function for the location parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword linear may be used to specify a linear part. If nothing is supplied, the location is taken to be constant unless the linear argument is given.

linear
A formula beginning with ~ in W&R notation, specifying the linear part of the regression function for the location parameter or list of two such expressions for the location and/or shape parameters.

pmu
Vector of initial estimates for the location parameters. If mu is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

pshape
If the link is Student t, an initial estimate of the degrees of freedom; if it is stable, an estimate of the tail parameter; if it is mixture, an estimate of the mixture probability.
wt              Weight vector.
envir           Environment in which model formulae are to be interpreted or a data object of class, repeated, tccov, or tvcov; the name of the response variable should be given in \( y \). If \( y \) has class repeated, it is used as the environment.
print.level    Arguments controlling \texttt{nlm}.
typsize         Arguments controlling \texttt{nlm}.
ndigit          Arguments controlling \texttt{nlm}.
gradtol         Arguments controlling \texttt{nlm}.
stepmax         Arguments controlling \texttt{nlm}.
steptol         Arguments controlling \texttt{nlm}.
iterlim         Arguments controlling \texttt{nlm}.
fscale          Arguments controlling \texttt{nlm}.

Details

Nonlinear regression models can be supplied as formulae where parameters are unknowns in which case factor variables cannot be used and parameters must be scalars. (See \texttt{finterp}.)

The printed output includes the -log likelihood (not the deviance), the corresponding AIC, the maximum likelihood estimates, standard errors, and correlations.

Value

A list of class \texttt{gnlm} is returned that contains all of the relevant information calculated, including error codes.

Author(s)

J.K. Lindsey

See Also

\texttt{finterp, glm, gnlr, gnlr3}

Examples

```r
# assay to estimate LD50
y <- c(9,9,10,4,1,0,0)
y <- cbind(y,10-y)
dose <- log10(100/(2.686,2.020,1.520,1.143,0.860,0.647,0.486))

summary(glm(y~dose, family=binomial))
bnlr(y, mu=dose, pmu=c(1,1))
summary(glm(y~dose, family=binomial(link=probit)))
bnlr(y, link="probit", mu=dose, pmu=c(1,1))
## Not run:
bnlr(y, link="log log", mu=dose, pmu=c(1,1))
bnlr(y, link="comp log log", mu=dose, pmu=c(1,1))
```
fit.dist fits the distributions in Chapter 4 of Lindsey (1995, 2003 2nd edn): binomial, beta-binomial, Poisson, negative binomial, geometric, zeta, normal, log normal, inverse Gauss, logistic, Laplace, Cauchy, Student t, exponential, Pareto, gamma, and Weibull to frequency (histogram) data, possibly plotting the frequency polygon of fitted values with the histogram.

**Usage**

```r
fit.dist(y, ni, distribution = "normal", breaks = FALSE, delta = 1,
         censor = FALSE, exact = TRUE, plot = FALSE, add = FALSE,
         xlab = deparse(substitute(y)), ylab = "Probability",
         xlim = range(y), main = paste("Histogram of",
         deparse(substitute(y))), ...)```

**Arguments**

- `y` Vector of observations.
- `ni` Corresponding vector of frequencies.
- `distribution` Character string specifying the distribution.
- `breaks` If TRUE, y contains breaks between categories instead of mid-points.
- `delta` Scalar or vector giving the unit of measurement (always one for discrete data) for each response value, set to unity by default. For example, if a response is measured to two decimals, delta=0.01.
- `censor` If TRUE, the last category is right censored.
- `exact` If FALSE, uses the approximations for certain distributions in Lindsey (1995).
- `plot` If TRUE, plots the histogram of observed frequencies and the frequency polygon of fitted values.
- `add` If TRUE, adds a new frequency polygon of fitted values without reploting the histogram.
- `xlab` Plotting control options.
Author(s)
J.K. Lindsey

References

Examples

f <- c(215, 1485, 5331, 10649, 14959, 11929, 6678, 2092, 342)
y <- seq(0, 8)
fit.dist(y, f, "binomial", plot=TRUE, xlab="Number of males", main="Distribution of males in families of 8 children")
#
f <- c(1, 1, 6, 3, 4, 3, 9, 6, 5, 16, 4, 11, 6, 11, 3, 4, 5, 6, 4, 4, 1, 1, 4, 1, 2,
0, 2, 0, 0, 1)
y <- seq(1100, 4100, by=100)
fit.dist(y, f, "normal", delta=100, plot=TRUE, xlab="Monthly salary (dollars)", main="Distribution of women mathematicians' salaries")
fit.dist(y, f, "log normal", delta=100, plot=TRUE, add=TRUE, lty=3)
fit.dist(y, f, "logistic", delta=100, exact=FALSE, plot=TRUE, add=TRUE, lty=2)

Description

fmr fits user specified nonlinear regression equations to the location parameter of the common one and two parameter distributions. (The log of the scale parameter is estimated to ensure positivity.)

Usage

fmr(y = NULL, distribution = "normal", mu = NULL, mix = NULL,
    linear = NULL, pmu = NULL, pmix = NULL, pshape = NULL,
censor = "right", exact = FALSE, wt = 1, delta = 1,
    common = FALSE, envir = parent.frame(), print.level = 0,
    typsize = abs(p), ndigit = 10, gradtol = 1e-05, stepmax = 10 *
    sqrt(p %*% p), steptol = 1e-05, iterlim = 100, fscale = 1)
Arguments

y
A response vector for uncensored data, a two column matrix for binomial data or censored data, with the second column being the censoring indicator (1: uncensored, 0: right censored, -1: left censored), or an object of class, response (created by restovec) or repeated (created by rmna or lvna). If the repeated data object contains more than one response variable, give that object in envir and give the name of the response variable to be used here.

distribution
Either a character string containing the name of the distribution or a function giving the -log likelihood and calling the location and mixture functions. Distributions are binomial, beta binomial, double binomial, multiplicative binomial, Poisson, negative binomial, double Poisson, multiplicative Poisson, gamma count, Consul, geometric, normal, inverse Gauss, logistic, exponential, gamma, Weibull, extreme value, Pareto, Cauchy, Student t, Laplace, and Levy. (For definitions of distributions, see the corresponding [dpqr]distribution help.)

mu
A user-specified function of pmu, and possibly linear, giving the regression equation for the location. This may contain a linear part as the second argument to the function. It may also be a formula beginning with ~, specifying either a linear regression function for the location parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword linear may be used to specify a linear part. If nothing is supplied, the location is taken to be constant unless the linear argument is given.

mix
A user-specified function of pmix, and possibly linear, giving the regression equation for the mixture parameter. This may contain a linear part as the second argument to the function. It may also be a formula beginning with ~, specifying either a linear regression function for the mixture parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword linear may be used to specify a linear part. If nothing is supplied, this parameter is taken to be constant. This parameter is the logit of the mixture probability.

linear
A formula beginning with ~ in W&R notation, or list of two such expressions, specifying the linear part of the regression function for the location or location and mixture parameters.

pmu
Vector of initial estimates for the location parameters. If mu is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

pmix
Vector of initial estimates for the mixture parameters. If mix is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

pshape
An initial estimate for the shape parameter.

censor
right, left, or both indicating where the mixing distribution is placed. both is only possible for binomial data.

exact
If TRUE, fits the exact likelihood function for continuous data by integration over intervals of observation given in delta, i.e. interval censoring.

wt
Weight vector.
delta  Scalar or vector giving the unit of measurement (always one for discrete data) for each response value, set to unity by default - for example, if a response is measured to two decimals, delta=0.01. If the response is transformed, this must be multiplied by the Jacobian. The transformation cannot contain unknown parameters. For example, with a log transformation, delta=1/y.

common  If TRUE, mu and mix must both be either functions with, as argument, a vector of parameters having some or all elements in common between them so that indexing is in common between them or formulae with unknowns. All parameter estimates must be supplied in pmu. If FALSE, parameters are distinct between the two functions and indexing starts at one in each function.

envir  Environment in which model formulae are to be interpreted or a data object of class, repeated, tccov, or tvcov; the name of the response variable should be given in y. If y has class repeated, it is used as the environment.

print.level  Arguments controlling nlm.

typsize  Arguments controlling nlm.

ndigit  Arguments controlling nlm.

gradtol  Arguments controlling nlm.

stepmax  Arguments controlling nlm.

steptol  Arguments controlling nlm.

iterlim  Arguments controlling nlm.

fscale  Arguments controlling nlm.

Details

For the Poisson and related distributions, the mixture involves the zero category. For the binomial and related distributions, it involves the two extreme categories. For all other distributions, it involves either left or right censored individuals. A user-specified -log likelihood can also be supplied for the distribution.

Nonlinear regression models can be supplied as formulae where parameters are unknowns in which case factor variables cannot be used and parameters must be scalars. (See finterp.)

The printed output includes the -log likelihood (not the deviance), the corresponding AIC, the maximum likelihood estimates, standard errors, and correlations.

Value

A list of class gnlm is returned that contains all of the relevant information calculated, including error codes.

Author(s)

J.K. Lindsey

See Also

finterp, glm, gnlr, gnlr3, lm.
Examples

```r
sex <- c(rep(0,10), rep(1,10))
age <- c(8,10,12,12,8,7,16,7,9,11,8,9,14,12,12,11,7,7,7,12)
y <- cbind(c(.2, .7, .3, 1.3, .2, .5, .9, .4, 1.9, .4, .8, .6, 3.3, 1.7, .2, 1.4, .7, .5, 0, .1),
          c(0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1))
# y <- cbind(rweibull(20,2,2+2*sex+age), rbinom(20,1,0.7))
# log linear regression with Weibull distribution with a point mass
# for right censored individuals
mu <- function(p) exp(p[1]+p[2]*sex+p[3]*age)
fmr(y, dist="Weibull", mu=mu, pmu=exp(1), pshape=1, pmix=0.5)
# or equivalently
fmr(y, dist="Weibull", mu=function(p, linear) exp(linear),
    linear=-sex+age, pmu=exp(1), pmix=0.5, pshape=1)
# or
fmr(y, dist="Weibull", mu=-exp(b0+b1*sex+b2*age), pmu=list(b0=4,b1=0,b2=0),
    pmix=0.5, pshape=1)
# include logistic regression for the mixture parameter
mix <- function(p) p[1]+p[2]*sex
fmr(y, dist="Weibull", mu=exp(a+age), mix=mix, pmu=c(4,0),
    pmix=c(10,0), pshape=0.5)
# or equivalently
fmr(y, dist="Weibull", mu=function(p, linear) exp(linear),
    linear=list(~age, ~sex), pmu=c(4,0), pmix=c(10,0), pshape=0.5)
# or
fmr(y, dist="Weibull", mu=-exp(b0+b1*age), mix=-c0+c1*sex,
    pmu=list(b0=4,b1=0), pmix=list(c0=10,c1=0), pshape=0.5)
# generate zero-inflated negative binomial data
x1 <- rpois(50,4)
x2 <- rpois(50,4)
ind <- rbinom(50,1/(1+exp(-1-0.1*x1))
y <- ifelse(ind, rbinom(50,3, mu=exp(1+0.2*x2)), 0)
# standard Poisson models
gnlr(y, dist="Poisson", mu=exp(a), pmu=1)
gnlr(y, dist="Poisson", mu=exp(linear), linear=-x2, pmu=c(1,0.2))
# zero-inflated Poisson ZIP
fmr(y, dist="Poisson", mu=exp(a), pmu=1, pmix=0)
fmr(y, dist="Poisson", mu=exp(linear), linear=-x2, pmu=c(1,0.2), pmix=0)
fmr(y, dist="Poisson", mu=exp(a), mix=-x1, pmu=1, pmix=c(1,0))
fmr(y, dist="Poisson", mu=exp(linear), linear=-x2, mix=-x1, pmu=c(1,0.2),
    pmix=c(1,0))
# zero-inflated negative binomial
fmr(y, dist="negative binomial", mu=exp(a), pmu=1, pshape=0, pmix=0)
fmr(y, dist="negative binomial", mu=exp(linear), linear=-x2, pmu=c(1,0.2),
    pshape=0, pmix=0)
fmr(y, dist="negative binomial", mu=exp(a), mix=-x1, pmu=1, pshape=0,
    pmix=c(1,0))
fmr(y, dist="negative binomial", mu=exp(linear), linear=-x2, mix=-x1,
    pshape=0, pmix=c(1,0))
```
Description

gnlr fits user-specified nonlinear regression equations to one or both parameters of the common one and two parameter distributions. A user-specified -log likelihood can also be supplied for the distribution. Most distributions allow data to be left, right, and/or interval censored.

Usage

gnlr(y = NULL, distribution = "normal", pmu = NULL, pshape = NULL, mu = NULL, shape = NULL, linear = NULL, exact = FALSE, wt = 1, delta = 1, shfn = FALSE, common = FALSE, envir = parent.frame(), print.level = 0, typsize = abs(p), ndigit = 10, gradtol = 1e-05, steptol = 1e-05, iterlim = 100, fscale = 1)

Arguments

- **y** A response vector for uncensored data, a two column matrix for binomial data or censored data, with the second column being the censoring indicator (1: uncensored, 0: right censored, -1: left censored), or an object of class, response (created by restovec) or repeated (created by rmna or lvna). If the repeated data object contains more than one response variable, give that object in envir and give the name of the response variable to be used here. The beta, simplex, and two-sided power distributions for proportions do not allow left or right censoring (only interval censoring).

- **distribution** Either a character string containing the name of the distribution or a function giving the -log likelihood. (In the latter case, all initial parameter estimates are supplied in pmu.) Distributions are binomial, beta binomial, double binomial, mult(iplicative) binomial, Poisson, negative binomial, double Poisson, mult(iplicative) Poisson, gamma count, Consul generalized Poisson, logarithmic series, geometric, normal, inverse Gauss, logistic, exponential, gamma, Weibull, extreme value, Cauchy, Pareto, Laplace, Levy, beta, simplex, and two-sided power. All but the binomial-based distributions and the beta, simplex, and two-sided power distributions may be right and/or left censored. (For definitions of distributions, see the corresponding [dpqr]distribution help.)

- **pmu** Vector of initial estimates for the location parameters. If mu is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list. If distribution is a user-supplied -log likelihood function, all initial estimates must be supplied here.
pshape Vector of initial estimates for the shape parameters. If shape is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

mu A user-specified function of pmu, and possibly linear, giving the regression equation for the location. This may contain a linear part as the second argument to the function. It may also be a formula beginning with ~, specifying either a linear regression function for the location parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword linear may be used to specify a linear part. If nothing is supplied, the location is taken to be constant unless the linear argument is given.

shape A user-specified function of pshape, and possibly linear and/or mu, giving the regression equation for the dispersion or shape parameter. This may contain a linear part as the second argument to the function and the location function as last argument (in which case shfn must be set to TRUE). It may also be a formula beginning with ~, specifying either a linear regression function for the shape parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword linear may be used to specify a linear part and the keyword mu to specify a function of the location parameter. If nothing is supplied, this parameter is taken to be constant unless the linear argument is given. This parameter is the logarithm of the usual one.

linear A formula beginning with ~ in W&R notation, specifying the linear part of the regression function for the location parameter or list of two such expressions for the location and/or shape parameters.

exact If TRUE, fits the exact likelihood function for continuous data by integration over intervals of observation given in delta, i.e. interval censoring.

wt Weight vector.

delta Scalar or vector giving the unit of measurement (always one for discrete data) for each response value, set to unity by default. For example, if a response is measured to two decimals, delta=0.01. If the response is transformed, this must be multiplied by the Jacobian. The transformation cannot contain unknown parameters. For example, with a log transformation, delta=1/y. (The delta values for the censored response are ignored.)

shfn If true, the supplied shape function depends on the location (function). The name of this location function must be the last argument of the shape function.

common If TRUE, mu and shape must both be either functions with, as argument, a vector of parameters having some or all elements in common between them so that indexing is in common between them or formulae with unknowns. All parameter estimates must be supplied in pmu. If FALSE, parameters are distinct between the two functions and indexing starts at one in each function.

envir Environment in which model formulae are to be interpreted or a data object of class, repeated, tccov, or tvcov; the name of the response variable should be given in y. If y has class repeated, it is used as the environment.

print.level Arguments controlling nlm.
Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Arguments controlling `nlm`.

Nonlinear regression models can be supplied as formulae where parameters are unknowns in which
case factor variables cannot be used and parameters must be scalars. (See `finterp`.)

The printed output includes the -log likelihood (not the deviance), the corresponding AIC, the max-
imum likelihood estimates, standard errors, and correlations.

A list of class `gnlm` is returned that contains all of the relevant information calculated, including
error codes.

J.K. Lindsey

`finterp`, `fmr`, `glm`, `gnlmix`, `glmm`, `gnlmm`, `gnlr3`, `lm`, `nlar`, `nls`.

sex <- c(rep(0,10),rep(1,10))
sexf <- gl(2,10)
age <- c(8,10,12,12,8,7,16,7,9,11,8,9,14,12,12,11,7,7,7,12)
y <- cbind(c(9.2, 7.3,13.0, 6.9, 3.9,14.9,17.8, 4.8, 6.4, 3.3,17.2,
14.4,17.0, 5.0,17.3, 3.8,19.4, 5.0, 2.0,19.0),
c(0,1,0,1,1,0,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1))
# y <- cbind(rweibull(20,2,2+2*sex+age),rbinom(20,1,0.7))
# linear regression with inverse Gauss distribution
gnlr(y, dist="inverse Gauss", mu=mu, pmu=c(3,0,0), pshape=1)
# or equivalently
gnlr(y, dist="inverse Gauss", mu=sexf+age, pmu=c(3,0,0), pshape=1)
# or
gnlr(y, dist="inverse Gauss", linear=sexf+age, pmu=c(3,0,0), pshape=1)
# or
gnlr(y, dist="inverse Gauss", mu=-b0+b1*sex+b2*age,
    pmu=list(b0=3,b1=0,b2=0), pshape=1)
# nonlinear regression with inverse Gauss distribution
mu <- function(p, linear) exp(linear)
gnlr(y, dist="inverse Gauss", mu=mu, linear=-sexf+age, pmu=c(3,0,0), pshape=-1)
# or equivalently
gnlr(y, dist="inverse Gauss", mu=exp(b0+b1*sex+b2*age), pmu=list(b0=3,b1=0,b2=0), pshape=-1)
# or

# include regression for the shape parameter with same mu function
gnlr(y, dist="inverse Gauss", mu=mu, linear=-sexf+age, shape=shape, pmu=c(3,0,0), pshape=c(3,0,0))
# or equivalently

gnlr(y, dist="inverse Gauss", mu=mu, linear=-sexf+age, shape=-sexf+age, pmu=c(3,0,0), pshape=c(3,0,0))
# or

gnlr(y, dist="inverse Gauss", mu=mu, linear=list(-sex+age,-sex+age), pmu=c(3,0,0), pshape=c(3,0,0))
# or

gnlr(y, dist="inverse Gauss", mu=mu, linear=-sex+age, shape=-c0+c1*sex+c2*age, pmu=c(3,0,0), pshape=list(c0=3,c1=0,c2=0))
# or

# shape as a function of the location
shape <- function(p, mu) p[1]+p[2]*sex+p[3]*mu
gnlr(y, dist="inverse Gauss", mu=age, shape=shape, pmu=c(3,0,0), pshape=c(3,0,0), shfn=TRUE)
# or

gnlr(y, dist="inverse Gauss", mu=age, shape=-a+b*sex+c*mu, pmu=c(3,0,0), pshape=c(3,0,0), shfn=TRUE)
# or

# common parameter in location and shape functions for age
mu <- function(p) exp(p[1]+p[2]*age)
gnlr(y, dist="inverse Gauss", mu=mu, shape=shape, pmu=c(3,0,1,0), common=TRUE)
# or

gnlr(y, dist="inverse Gauss", mu=exp(a+b*age), shape=-c+d*sex+b*age, pmu=c(3,0,1,0), common=TRUE)
# or

# user-supplied -log likelihood function
y <- rnorm(20,2+3*sex,2)
dist <- function(p)-sum(dnorm(y,p[1]+p[2]*sex,p[3],log=TRUE))
gnlr(y, dist=dist,pmu=1:3)
dist <- -sum(dnorm(y,a*b*sex,v,log=TRUE))
gnlr(y, dist=dist,pmu=1:3)
gnlr3  Generalized Nonlinear Regression Models for Three Parameter Distributions

Description

gnlr3 fits user specified nonlinear regression equations to one, two, or all three parameters of three parameter distributions. Continuous data may be left, right, and/or interval censored.

Usage

gnlr3(y = NULL, distribution = "normal", mu = NULL, shape = NULL, family = NULL, linear = NULL, pmu = NULL, pshape = NULL, pfamily = NULL, exact = FALSE, wt = 1, common = FALSE, delta = 1, envir = parent.frame(), print.level = 0, typsize = abs(p), ndigit = 10, gradtol = 1e-05, stepmax = 10 * sqrt(p %*% p), steptol = 1e-05, iterlim = 100, fscale = 1)

Arguments

y  The response vector for uncensored data, two columns for censored data, with the second being the censoring indicator (1: uncensored, 0: right censored, -1: left censored.), or an object of class, response (created by restovec or repeated (created by rmna or lvna). If the repeated data object contains more than one response variable, give that object in envir and give the name of the response variable to be used here.
distribution  Either a character string containing the name of the distribution or a function giving the -log likelihood and calling the location, shape, and family functions. Distributions are Box-Cox transformed normal, generalized inverse Gauss, generalized logistic, Hjorth, generalized gamma, Burr, generalized Weibull, power exponential, Student t, generalized extreme value, power variance function Poisson, and skew Laplace. (For definitions of distributions, see the corresponding [dpqr]distribution help.)
mu  A user-specified function of pmu, and possibly linear, giving the regression equation for the location. This may contain a linear part as the second argument to the function. It may also be a formula beginning with ~, specifying either a linear regression function for the location parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword linear may be used to specify a linear part. If nothing is supplied, the location is taken to be constant unless the linear argument is given.
shape  A user-specified function of pshape, and possibly linear, giving the regression equation for the dispersion or shape parameter. This may contain a linear part as the second argument to the function. It may also be a formula beginning with ~, specifying either a linear regression function for the shape parameter in the Wilkinson and Rogers notation or a general function with named unknown
parameters. If it contains unknown parameters, the keyword `linear` may be used to specify a linear part. If nothing is supplied, this parameter is taken to be constant unless the linear argument is given. This parameter is the logarithm of the usual one.

**family**  
A user-specified function of `pfamily`, and possibly `linear`, for the regression equation of the third (family) parameter of the distribution. This may contain a linear part that is the second argument to the function. It may also be a formula beginning with `~`, specifying either a linear regression function for the family parameter in the Wilkinson and Rogers notation or a general function with named unknown parameters. If neither is supplied, this parameter is taken to be constant unless the linear argument is given. In most cases, this parameter is the logarithm of the usual one.

**linear**  
A formula beginning with `~` in W&R notation, specifying the linear part of the regression function for the location parameters or list of three such expressions for the location, shape, and/or family parameters.

**pmu**  
Vector of initial estimates for the location parameters. If `mu` is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

**pshape**  
Vector of initial estimates for the shape parameters. If `shape` is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

**pfamily**  
Vector of initial estimates for the family parameters. If `family` is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

**exact**  
If TRUE, fits the exact likelihood function for continuous data by integration over intervals of observation given in `delta`, i.e. interval censoring.

**wt**  
Weight vector.

**common**  
If TRUE, at least two of `mu`, `shape`, and `family` must both be either functions with, as argument, a vector of parameters having some or all elements in common between them so that indexing is in common between them or formulae with unknowns. All parameter estimates must be supplied in `pmu`. If FALSE, parameters are distinct between the two functions and indexing starts at one in each function.

**delta**  
Scalar or vector giving the unit of measurement (always one for discrete data) for each response value, set to unity by default - for example, if a response is measured to two decimals, `delta=0.01`. If the response is transformed, this must be multiplied by the Jacobian. The transformation cannot contain unknown parameters. For example, with a log transformation, `delta=1/y`. (The delta values for the censored response are ignored.)

**envir**  
Environment in which model formulae are to be interpreted or a data object of class, `repeated`, `tccov`, or `tvcov`; the name of the response variable should be given in `y`. If `y` has class `repeated`, it is used as the environment.

**print.level**  
Arguments controlling `nlm`.

**typosize**  
Arguments controlling `nlm`.

**ndigit**  
Arguments controlling `nlm`.
gradtol  Arguments controlling nlm.
stepmax  Arguments controlling nlm.
steptol  Arguments controlling nlm.
iterlim  Arguments controlling nlm.
fscale   Arguments controlling nlm.

Details

Nonlinear regression models can be supplied as formulae where parameters are unknowns in which case factor variables cannot be used and parameters must be scalars. (See finterp.)

The printed output includes the -log likelihood (not the deviance), the corresponding AIC, the maximum likelihood estimates, standard errors, and correlations.

Value

A list of class gnlm is returned that contains all of the relevant information calculated, including error codes.

Author(s)

J.K. Lindsey

See Also

finterp, fmr, glm, gnlr, lm, nlr, nls.

Examples

sex <- c(rep(0,10),rep(1,10))
sexf <- gl(2,10)
age <- c(8,10,12,12,8,7,16,7,9,11,8,9,14,12,12,11,7,7,7,12)
y <- cbind(c(9.2, 7.3, 13.0, 6.9, 3.9, 14.9, 17.8, 4.8, 6.4, 3.3, 17.2,
            14.4, 17.0, 5.0, 17.3, 3.8, 19.4, 5.0, 2.0, 19.0),
            c(0,1,0,1,1,0,1,0,1,1,1,1,1,1,1,1,1,1,1,1))
# y <- cbind(rweibull(20,2,2+2*sex+age),rbinom(20,1,0.7))
# log linear regression with the generalized Weibull distribution
mu <- function(p) exp(p[1]+p[2]*sex+p[3]*age)
gnlr3(y, dist="Weibull", mu=mu, pmu=c(3,1,0), pshape=2, pfamily=-2)
# or equivalently
mul <- function(p,linear) exp(linear)
gnlr3(y, dist="Weibull", mu=mul, linear=sex+age, pmu=c(3,1,0),
     pshape=2, pfamily=-2)
# or
gnlr3(y, dist="Weibull", mu=exp(b0+b1*sex+b2*age),
     pmu=list(b0=3,b1=1,b2=0), pshape=2, pfamily=-2)
# include regression for the shape parameter with same mu function
gnlr3(y, dist="Weibull", mu=mu, shape=shape,
Nonlinear Normal, Gamma, and Inverse Gaussian Regression Models

Description

`nlr` fits a user-specified nonlinear regression equation by least squares (normal) or its generalization for the gamma and inverse Gauss distributions.

Usage

```r
nlr(y = NULL, mu = NULL, pmu = NULL, distribution = "normal", wt = 1, delta = 1, envir = parent.frame(), print.level = 0, typsize = abs(pmu), ndigit = 10, gradtol = 1e-05, stepmax = 10 * sqrt(pmu **%** pmu), steptol = 1e-05, iterlim = 100, fscale = 1)
```

Arguments

- **y**: The response vector or an object of class `response` (created by `restovec`) or `repeated` (created by `rmna` or `lvna`).
- **mu**: A function of \( p \) giving the regression equation for the mean or a formula beginning with `~`, specifying either a linear regression function in the Wilkinson and Rogers notation or a general nonlinear function with named unknown parameters.
- **pmu**: Vector of initial estimates of the parameters. If `mu` is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.
- **distribution**: The distribution to be used: normal, gamma, or inverse Gauss.
- **wt**: Weight vector.
- **delta**: Scalar or vector giving the unit of measurement for each response value, set to unity by default. For example, if a response is measured to two decimals, `delta = 0.01`. If the response is transformed, this must be multiplied by the Jacobian. For example, with a log transformation, `delta = 1/y`.
- **envir**: Environment in which model formulae are to be interpreted or a data object of class `repeated`, `tccov`, or `tvcov`. If `y` has class `repeated`, it is used as the environment.
- **print.level**: Arguments controlling `nlm`.
- **typsize**: Arguments controlling `nlm`.
- **ndigit**: Arguments controlling `nlm`.
- **gradtol**: Arguments controlling `nlm`.
- **stepmax**: Arguments controlling `nlm`.
- **steptol**: Arguments controlling `nlm`.
- **iterlim**: Arguments controlling `nlm`.
- **fscale**: Arguments controlling `nlm`.

Details

A nonlinear regression model can be supplied as a formula where parameters are unknowns in which case factor variables cannot be used and parameters must be scalars. (See `finterp`.)

The printed output includes the -log likelihood (not the deviance), the corresponding AIC, the parameter estimates, standard errors, and correlations.

Value

A list of class `n1r` is returned that contains all of the relevant information calculated, including error codes.

Author(s)

J.K. Lindsey
Nonlinear Ordinal Regression Models

nordr fits arbitrary nonlinear regression functions (with logistic link) to ordinal response data by proportional odds, continuation ratio, or adjacent categories.

Usage

nordr(y = NULL, distribution = "proportional", mu = NULL, linear = NULL, pmu = NULL, pintercept = NULL, weights = NULL, envir = parent.frame(), print.level = 0, ndigit = 10, gradtol = 1e-05, steptol = 1e-05, fscale = 1, iterlim = 100, typsize = abs(p), stepmax = 10 * sqrt(p **% p))
Arguments

\textbf{y} \hspace{1cm} A vector of ordinal responses, integers numbered from zero to one less than the number of categories or an object of class \texttt{response} (created by \texttt{restovec}) or \texttt{repeated} (created by \texttt{rmna}) or \texttt{lyna}). If the repeated data object contains more than one response variable, give that object in \texttt{envir} and give the name of the response variable to be used here.

\textbf{distribution} \hspace{1cm} The ordinal distribution: proportional odds, continuation ratio, or adjacent categories.

\textbf{mu} \hspace{1cm} User-specified function of pmu, and possibly linear, giving the logistic regression equation. This must contain the first intercept. It may contain a linear part as the second argument to the function. It may also be a formula beginning with \texttt{~}, specifying a logistic regression function for the location parameter, either a linear one using the Wilkinson and Rogers notation or a general function with named unknown parameters. If it contains unknown parameters, the keyword \texttt{linear} may be used to specify a linear part. If nothing is supplied, the location is taken to be constant unless the linear argument is given.

\textbf{linear} \hspace{1cm} A formula beginning with \texttt{~} in W&R notation, specifying the linear part of the logistic regression function.

\textbf{pmu} \hspace{1cm} Vector of initial estimates for the regression parameters, including the first intercept. If \texttt{mu} is a formula with unknown parameters, their estimates must be supplied either in their order of appearance in the expression or in a named list.

\textbf{pintercept} \hspace{1cm} Vector of initial estimates for the contrasts with the first intercept parameter (difference in intercept for successive categories): two less than the number of different ordinal values.

\textbf{weights} \hspace{1cm} Weight vector for use with contingency tables.

\textbf{envir} \hspace{1cm} Environment in which model formulae are to be interpreted or a data object of class \texttt{repeated}, \texttt{tccov}, or \texttt{tvcov}; the name of the response variable should be given in \texttt{y}. If \texttt{y} has class \texttt{repeated}, it is used as the environment.

\textbf{print.level} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{ndigit} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{gradtol} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{steptol} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{fscale} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{iterlim} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{typsize} \hspace{1cm} Arguments controlling \texttt{nlm}.

\textbf{stepmax} \hspace{1cm} Arguments controlling \texttt{nlm}.

Details

Nonlinear regression models can be supplied as formulae where parameters are unknowns in which case factor variables cannot be used and parameters must be scalars. (See \texttt{finterp}.)

The printed output includes the -log likelihood (not the deviance), the corresponding AIC, the maximum likelihood estimates, standard errors, and correlations.
Value

A list of class `nordr` is returned that contains all of the relevant information calculated, including error codes.

Author(s)

J.K. Lindsey

See Also

`finterp`, `fmr`, `glm`, `glm`, `gnlmm`, `gnlr`, `gnlr3`, `nlr`, `ordglm`

Examples

```r
# McCullagh (1980) JRSS B42, 109-142
# tonsil size: 2x3 contingency table
y <- c(0:2,0:2)
carrier <- c(rep(0,3),rep(1,3))
carrierf <- gl(2,3,6)
wt <- c(19,29,24, 497,560,269)
mu <- c(-1,0.5)
mu <- function(p) c(rep(p[1],3),rep(p[1]+p[2],3))
# proportional odds
# with mean function
nordr(y, dist="prop", mu=mu, pmu=pmu, weights=wt, pintercept=1.5)
# using Wilkinson and Rogers notation
nordr(y, dist="prop", mu=-carrierf, pmu=pmu, weights=wt, pintercept=1.5)
# using formula with unknowns
nordr(y, dist="prop", mu=-b0+b1*carrier, pmu=pmu, weights=wt, pintercept=1.5)
# continuation ratio
nordr(y, dist="cont", mu=mu, pmu=pmu, weights=wt, pintercept=1.5)
# adjacent categories
nordr(y, dist="adj", mu=-carrierf, pmu=pmu, weights=wt, pintercept=1.5)

# Haberman (1974) Biometrics 30, 589-600
# institutionalized schizophrenics: 3x3 contingency table
y <- rep(0:2,3)
fr <- c(43,6,9, 16,11,18, 3,10,16)
length <- gl(3,3)
## Not run:
# fit continuation ratio model with nordr and as a logistic model
nordr(y, mu=length, weights=fr, pmu=c(0,-1.4,-2.3), pint=0.13, dist="cont")

## End(Not run)
# logistic regression with reconstructed table
frcr <- cbind(c(43,16,3,49,27,13),c(6,11,10,9,18,16))
lengthord <- gl(3,1,6)
```
block <- gl(2,3)
summary(glm(frcr~lengthord+block,fam=binomial))
# note that AICs and deviances are different

ordglm

Generalized Linear Ordinal Regression Models

Description

ordglm fits linear regression functions with logistic or probit link to ordinal response data by proportional odds.

Usage

ordglm(formula, data = parent.frame(), link = "logit", maxiter = 10,
weights = 1)

Arguments

  formula  A model formula. The response must be integers numbered from zero to one
             less than the number of ordered categories.
  data     An optional data frame containing the variables in the model.
  link     Logit or probit link function.
  maxiter  Maximum number of iterations allowed.
  weights  A vector containing the frequencies for grouped data.

Value

A list of class ordglm is returned. The printed output includes the -log likelihood, the corresponding
AIC, the deviance, the maximum likelihood estimates, standard errors, and correlations.

Author(s)

J.K. Lindsey, adapted and heavily modified from Matlab code (ordinalMLE) by J.H. Albert.

References


See Also

glm, nordr
Examples

# McCullagh (1980) JRSS B42, 109-142
# tonsil size: 2x3 contingency table
y <- c(0:2,0:2)
carrier <- gl(2,3,6)
wt <- c(19,29,24,497,560,269)
ordglm(y~carrier, weights=wt)

rs2

Two-factor Box-Tidwell Nonlinear Response Surface Models

Description

rs2 fits a two-covariate power-transformed response surface by iterating the function, glm.

Usage

rs2(y, x1, x2, power = c(1, 1), weight = rep(1, length(x1)),
    family = gaussian, iterlim = 20)

Arguments

y       Response variable
x1      First covariate
x2      Second covariate
power   Initial estimates of the two power transformations
weight  Weight vector
family  glm family
iterlim Iteration limit

Value

A list of class, rs, is returned containing the model and the power estimates.

Author(s)

J.K. Lindsey

See Also

lm, glm, gnlr, gnlr3, rs3
Examples

```r
x1 <- rep(1:4,5)
x2 <- rep(1:5,rep(4,5))
y <- rpois(20,1+2*sqrt(x1)+3*log(x2)+4*x1+log(x2)^2+2*sqrt(x1)*log(x2))
rs2(y, x1, x2, family=poisson)
```

---

**rs3**  
*Three-factor Box-Tidwell Nonlinear Response Surface Models*

**Description**

rs3 fits a three-covariate power-transformed response surface by iterating the function, `glm`.

**Usage**

```r
rs3(y, x1, x2, x3, power = c(1, 1, 1), weight = rep(1, length(x1)),
    family = gaussian, iterlim = 20)
```

**Arguments**

- `y`: Response variable
- `x1`: First covariate
- `x2`: Second covariate
- `x3`: Third covariate
- `power`: Initial estimates of the three power transformations
- `weight`: Weight vector
- `family`: glm family
- `iterlim`: Iteration limit

**Value**

A list of class, `rs`, is returned containing the model and the power estimates.

**Author(s)**

J.K. Lindsey

**See Also**

`lm, glm, gnlr, gnlr3, rs2`
Examples

x1 <- rep(1:4, 5)
x2 <- rep(1:5, rep(4, 5))
x3 <- c(rep(1:3, 6), 1, 2)

# y <- rpois(20, 1 + 2 * sqrt(x1) + 3 * log(x2) + 1/x3 + 4 * x1 + log(x2) ^ 2 + 1/x3 ^ 2 +
# 2 * sqrt(x1) * log(x2) + sqrt(x1) / x3 + 1 * log(x2) / x3)
y <- c(9, 11, 14, 33, 11, 20, 27, 22, 32, 24, 20, 28, 25, 41, 26, 31, 37, 34)
rs3(y, x1, x2, x3, family=poisson)
Index

*Topic models
  bnlr, 2
  fit.dist, 4
  fmr, 5
  gnlr, 9
  gnlr3, 13
  nlr, 16
  nordr, 18
  ordglm, 21
  rs2, 22
  rs3, 23

bnlr, 2

finterp, 3, 7, 11, 15, 17–20
fit.dist, 4
fmr, 5, 11, 15, 18, 20

glm, 3, 7, 11, 15, 18, 20–23
glmm, 11, 18, 20
gnlmm, 11
gnlrm, 11, 18, 20
gnlr, 3, 7, 9, 15, 18, 20, 22, 23
gnlr3, 3, 7, 11, 13, 18, 20, 22, 23

lm, 7, 11, 15, 18, 22, 23
lvna, 2, 6, 9, 13, 17, 19

nlm, 3, 7, 10, 11, 14, 15, 17, 19
nlr, 11, 15, 16, 20
nls, 11, 15, 18
nordr, 18, 21

ordglm, 20, 21

restovec, 2, 6, 9, 13, 17, 19
rmna, 2, 6, 9, 13, 17, 19
rs2, 22, 23
rs3, 22, 23