Package ‘rregm’

January 28, 2026

Type Package

Title Reparameterized Regression Models

Version 1.0

Maintainer Diego Gallardo <`dgallardo@ubiobio.cl`>

Description Provides estimation and data generation tools for several new regression models, including the gamma, beta, inverse gamma and beta prime distributions. These models can be parameterized based on the mean, median, mode, geometric mean and harmonic mean, as specified by the user.
For details, see Bourguignon and Gallardo (2025a) <doi:10.1016/j.chemolab.2025.105382> and Bourguignon and Gallardo (2025b) <doi:10.1111/stan.70007>.

Depends R (>= 4.0.0), stats

Imports extraDistr, pracma, gamlss, gamlss.dist, invgamma

License GPL (>= 2)

Encoding UTF-8

NeedsCompilation no

Author Diego Gallardo [aut, cre],
Marcelo Bourguignon [aut],
Marcia Brandao [aut]

Repository CRAN

Date/Publication 2026-01-28 19:10:02 UTC

Contents

RBE *Tools for a reparameterized beta regression model*

Description

A set of functions related to the reparameterized beta regression model based on different measures of central tendency: mean, median, mode, geometric mean or harmonic mean.

Usage

```
BEAM(mu.link = "logit", sigma.link = "log")
BEGM(mu.link = "logit", sigma.link = "log")
BEHM(mu.link = "logit", sigma.link = "log")
BEMD(mu.link = "logit", sigma.link = "log")
BEMO(mu.link = "logit", sigma.link = "log")
dBEAM(x, mu = 0.5, sigma = 1, log = FALSE)
dBEGM(x, mu = 0.5, sigma = 1, log = FALSE)
dBEHM(x, mu = 0.5, sigma = 1, log = FALSE)
dBEMD(x, mu = 0.5, sigma = 1, log = FALSE)
dBEMO(x, mu = 0.5, sigma = 1, log = FALSE)
dRBE(x, mu=0.5, sigma=1, param="AM", log=FALSE)
fit.RBE(formula = formula(data), sigma.formula=~1, data, param="AM")
pBEAM(q, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pBEGM(q, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pBEHM(q, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pBEMD(q, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pBEMO(q, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pRBE(q, mu=0.5, sigma=1, param="AM", lower.tail = TRUE, log.p = FALSE)
qBEAM(p, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qBEGM(p, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qBEHM(p, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qBEMD(p, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qBEMO(p, mu = 0.5, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qRBE(p, mu=0.5, sigma=1, param="AM", lower.tail = TRUE, log.p = FALSE)
rBEAM(n, mu = 0.5, sigma = 1)
rBEGM(n, mu = 0.5, sigma = 1)
rBEHM(n, mu = 0.5, sigma = 1)
rBEMD(n, mu = 0.5, sigma = 1)
rBEMO(n, mu = 0.5, sigma = 1)
rRBE(n, mu=0.5, sigma=1, param="AM")
```

Arguments

`mu.link`	the mu link function with default logit
`sigma.link`	the sigma link function with default log
`mu, sigma`	vector of parameter values

`formula`	an object of class "*formula*" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`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), typically the environment from which lm is called.
`sigma.formula`	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. sigma.formula=~x1+x2.
`param`	parameterization used for the model. "AM" for mean, "MD" for median, "MO" for mode, "GM" for geometric mean, and "HM" for harmonic mean.
`x, q`	vector of quantiles
`p`	vector of probabilities
`n`	number of observations. If length$(n) > 1$, the length is taken to be the number required.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE, probabilities are $P(X \leq x)$ otherwise, $P(X > x)$.

Details

The parameterization for the reparameterized beta distribution is given by

$$f(x;\mu,\sigma) = \frac{x^{\mu\,\sigma+\tau_1-1}(1-x)^{(1-\mu)\sigma+\tau_2-\tau_1-1}}{B(\mu\,\sigma+\tau_1,(1-\mu)\sigma+\tau_2-\tau_1)}, \quad 0 < x < 1,$$

where $0 < \mu < 1$, $\sigma > 0$ and τ_1 and τ_2 are constant. The following cases are highlighted:

- param="AM": $\tau_1 = \tau_2 = 0$ and μ represents the mean of the distribution.

- param="GM": $\tau_1 = \tau_2 = 1/2$ and μ represents the geometric mean of the distribution.

- param="HM": $\tau_1 = \tau_2 = 1$ and μ represents the harmonic mean of the distribution.

- param="MO": $\tau_1 = 1$ and $\tau_2 = 2$ and μ represents the mode of the distribution.

- param="MD": $\tau_1 = 1/2$ and $\tau_2 = 0$ and μ represents the median of the distribution.

Suppose the central tendency and the concentration parameter of Y_i satisfies the following functional relations

$$\text{logit}(\mu_i) = \mathbf{x}_i^\top \boldsymbol{\xi} \quad \text{and} \quad \log(\sigma_i) = \eta_{2i} = \mathbf{z}_i^\top \boldsymbol{\nu},$$

where $\text{logit}(u) = \log(u/(1-u))$ is the logit function, $\boldsymbol{\xi} = (\xi_1,\ldots,\xi_p)^\top$ and $\boldsymbol{\nu} = (\nu_1,\ldots,\nu_q)^\top$ are vectors of unknown regression coefficients which are assumed to be functionally independent, $\boldsymbol{\xi} \in \mathbb{R}^p$ and $\boldsymbol{\nu} \in \mathbb{R}^q$, with $p + q < n$, and $\mathbf{x}_i = (x_{i1},\ldots,x_{ip})^\top$ and $\mathbf{z}_i = (z_{i1},\ldots,z_{iq})^\top$ are observations on p and q known regressors, for $i = 1,\ldots,n$. Furthermore, we assume that the covariate matrices $\mathbf{X} = (\mathbf{x}_1,\ldots,\mathbf{x}_n)^\top$ and $\mathbf{Z} = (\mathbf{z}_1,\ldots,\mathbf{z}_n)^\top$ have rank p and q, respectively.

For this model, the Pearson's residuals are given by

$$r_i = \frac{y_i - m_i}{s_i}, \quad i = 1,\ldots,n,$$

where

$$m_i = \frac{\mu_i\sigma_i + \tau_1}{\sigma_i + \tau_2} \quad \text{and} \quad s_i = \sqrt{\frac{(\mu_i\sigma_i + \tau_1)((1-\mu_i)\sigma_i + \tau_2 - \tau_1)}{(\sigma_i + \tau_2)^2(\sigma_i + \tau_2 + 1)}}.$$

whereas the modified Pearson's residuals are given by

$$r_i^* = \frac{\text{logit}(y_i) - m_i^*}{s_i^*}, \quad i = 1, \ldots, n,$$

where

$$m_i^* = \psi(\mu_i\sigma_i + \tau_1) - \psi((1-\mu_i)\sigma_i + \tau_2 - \tau_1) \quad \text{and} \quad s_i^* = \sqrt{\psi'(\mu_i\sigma_i + \tau_1) + \psi'((1-\mu_i)\sigma_i + \tau_2 - \tau_1)},$$

with $\psi(\cdot)$ and $\psi'(\cdot)$ denoting the digamma and trigamma functions, respectively. Finally, the quantile residuals are given by

$$r_i^q = \Phi^{-1}\left(I_{y_i}(\mu_i\sigma_i + \tau_1, (1-\mu_i)\sigma_i + \tau_2 - \tau_1)\right), \quad i = 1, \ldots, n,$$

where $\Phi^{-1}(\cdot)$ denotes the inverse of the cumulative distribution function for the standard normal model and $I_y(\alpha, \beta) = B_x(\alpha, \beta)/B(\alpha, \beta)$ is the incomplete beta function ratio, $B_x(\alpha, \beta) = \int_0^x \omega^{\alpha-1}(1-\omega)^{\beta-1}\mathrm{d}\omega$ is the incomplete beta function, $B(\alpha, \beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta)$ is the beta function and $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1}\mathrm{e}^{-\omega}\mathrm{d}\omega$ is the gamma function. dRBE gives the density, pRBE gives the distribution function, qRBE gives the quantile function, and rRBE generates random deviates from the beta distribution with the specified parameterization. In addition, dBEXX, pBEXX, qBEXX and rBEXX also provides the equivalent functions for a specified parameterization for XX: AM (mean), GM (geometric mean), HM (harmonic mean), MD (median) and MO (mode). For instance, dBEAM gives the density for the beta model parameterized in the mean, pBEGM gives the distribution function for the beta model parameterized in the geometric mean and so on. Finally, the functions BEAM, BEGM, BEHM, BEMD and BEMO also provide a framework to fit models with *gamlss*.

Value

an object of class "rregm" is returned. The object returned for this functions is a list containing the following components:

`estimate`	A matrix containing the estimates and standard errors.
`logLik`	the log-likelihood function evaluated at the corresponding estimators.
`AIC`	the Akaike information criterion.
`BIC`	the Bayesian information criterion.
`tau1, tau2`	values for tau1 and tau2, depending on the considered parameterization.
`pearson.res`	Pearson's residuals.
`mod.pearson.res`	modified Pearson's residuals.
`quant.res`	quantile residuals.
`convergence`	logical. If convergence was attained.
`dist`	BE (the beta distribution).
`param`	The specified parameterization.
`mu.x`	design matrix for mu.
`sigma.x`	design matrix for sigma.

Author(s)

Diego Gallardo and Marcelo Bourguignon.

References

Bourguignon, M., Gallardo, D.I. (2025) A general and unified parameterization of the beta distribution: A flexible and robust beta regression model. Statistica Neerlandica, 79(2), e70007.

Examples

```
set.seed(2100)
n=100; x1=rnorm(max(n)) ##drawing covariates, the same for mu and sigma
mu=plogis(0.5-0.4*x1); sigma=exp(-0.1+0.05*x1)
y=rRBE(n, mu, sigma, param="MD") ## model parameterized in the median
data=list(y=y, x1=x1)
aux.RBE=fit.RBE(y~x1, sigma.formula=~x1, data=data, param="MD")
summary(aux.RBE)
qqnorm(res(aux.RBE, type="mod.pearson"))
#The beta model parameterized in the median also can be fitted using gamlss
#gamlss(y~x1, sigma.formula=~x1, data=data, family=BEMD)
```

`RBP` *Tools for a reparameterized beta prime regression model*

Description

A set of functions related to the reparameterized beta prime regression model based on different measures of central tendency: mean, median, mode, geometric mean or harmonic mean.

Usage

```
fit.RBP(formula = formula(data), sigma.formula=~1, data, param="AM")
dRBP(x, mu=1, sigma=1.5, param="AM", log=FALSE)
pRBP(q, mu=1, sigma=1.5, param="AM", lower.tail = TRUE, log.p = FALSE)
qRBP(p, mu=1, sigma=1.5, param="AM", lower.tail = TRUE, log.p = FALSE)
rRBP(n, mu=1, sigma=1.5, param="AM")
```

Arguments

`mu, sigma`	vector of parameter values
`formula`	an object of class "*formula*" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`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), typically the environment from which lm is called.

`sigma.formula`	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. sigma.formula=~x1+x2.
`param`	parameterization used for the model. "AM" for mean, "MD" for median, "MO" for mode, "GM" for geometric mean, and "HM" for harmonic mean.
`x, q`	vector of quantiles
`p`	vector of probabilities
`n`	number of observations. If length$(n) > 1$, the length is taken to be the number required.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE, probabilities are $P(X \leq x)$ otherwise, $P(X > x)$.

Details

The parameterization for the reparameterized beta prime distribution is given by

$$f(x;\mu,\sigma) = \frac{x^{\mu\,\sigma+\tau_1-1}(1+x)^{-(\sigma(1+\mu)+\tau_1-\tau_2+1)}}{B(\mu\,\sigma+\tau_1,\sigma-\tau_2+1)}, \quad 0 < x < 1,$$

where $\delta = \delta(\sigma) = (\sqrt{\sigma(\sigma+4)}+\sigma)/2$, $0 < \mu < 1$, $\sigma > 0$ and τ is a constant. The following cases are highlighted:

- param="AM": $\tau = 0$ and μ represents the mean of the distribution.

- param="GM": $\tau = 1/2$ and μ represents the geometric mean of the distribution.

- param="MD": $\tau = 1/3$ and μ represents the median of the distribution.

- param="MO" or ="HM": $\tau = 1$ and μ represents the mode or the harmonic mean of the distribution.

Suppose the central tendency and the concentration parameter of Y_i satisfies the following functional relations

$$\log(\mu_i) = \mathbf{x}_i^\top \boldsymbol{\xi} \quad \text{and} \quad \log(\sigma_i) = \eta_{2i} = \mathbf{z}_i^\top \boldsymbol{\nu},$$

where $\boldsymbol{\xi} = (\xi_1,\ldots,\xi_p)^\top$ and $\boldsymbol{\nu} = (\nu_1,\ldots,\nu_q)^\top$ are vectors of unknown regression coefficients which are assumed to be functionally independent, $\boldsymbol{\xi} \in \mathbb{R}^p$ and $\boldsymbol{\nu} \in \mathbb{R}^q$, with $p + q < n$, and $\mathbf{x}_i = (x_{i1},\ldots,x_{ip})^\top$ and $\mathbf{z}_i = (z_{i1},\ldots,z_{iq})^\top$ are observations on p and q known regressors, for $i = 1,\ldots,n$. Furthermore, we assume that the covariate matrices $\mathbf{X} = (\mathbf{x}_1,\ldots,\mathbf{x}_n)^\top$ and $\mathbf{Z} = (\mathbf{z}_1,\ldots,\mathbf{z}_n)^\top$ have rank p and q, respectively.

For this model, the Pearson's residuals are given by

$$r_i = \frac{y_i - m_i}{s_i}, \quad i = 1,\ldots,n,$$

where

$$m_i = \frac{\mu_i\sigma_i + \tau_1}{\sigma_i - \tau_2 + 1} \quad \text{and} \quad s_i = \frac{(\mu_i\sigma_i + \tau_1)((1+\mu_i)\sigma_i + \tau_1 - \tau_2 + 1)}{(\sigma_i + \tau - 1)}(\sigma_i - \tau_2)^{-1/2}.$$

Note that the Pearson's residuals are well defined as long as $\sigma_i - \tau_2 > 1$, $\forall i = 1,\ldots,n$. On the other hand, the modified Pearson's residuals are given by

$$r_i^* = \frac{\text{logit}(y_i) - m_i^*}{s_i^*}, \quad i = 1,\ldots,n,$$

where

$$m_i^* = \psi(\mu_i\sigma_i+\tau_1)-\psi((1+\mu_i)\sigma_i+\tau_1-\tau_2+1) \quad \text{and} \quad s_i^* = \sqrt{\psi'(\mu_i\sigma_i+\tau_1)-\psi'((1+\mu_i)\sigma_i+\tau_1-\tau_2+1)},$$

with $\psi(\cdot)$ and $\psi'(\cdot)$ denoting the digamma and trigamma functions, respectively. Finally, the quantile residuals are given by

$$r_i^q = \Phi^{-1}\left(I_{\frac{y_i}{1+y_i}}(\mu_i\sigma_i+\tau_1,(1+\mu_i)\sigma_i+\tau_1-\tau_2+1)\right), \quad i=1,\dots,n,$$

where $\Phi^{-1}(\cdot)$ denotes the inverse of the cumulative distribution function for the standard normal model and $I_y(\alpha,\beta)=B_x(\alpha,\beta)/B(\alpha,\beta)$ is the incomplete beta function ratio, $B_x(\alpha,\beta)=\int_0^x \omega^{\alpha-1}(1-\omega)^{\beta-1}\mathrm{d}\omega$ is the incomplete beta function, $B(\alpha,\beta)=\Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta)$ is the beta function and $\Gamma(\alpha)=\int_0^\infty \omega^{\alpha-1}\mathrm{e}^{-\omega}\mathrm{d}\omega$ is the gamma function. dRBP gives the density, pRBP gives the distribution function, qRBP gives the quantile function, and rRBP generates random deviates from the beta distribution with the specified parameterization.

Value

an object of class "rregm" is returned. The object returned for this functions is a list containing the following components:

`estimate`	A matrix containing the estimates and standard errors.
`logLik`	the log-likelihood function evaluated at the corresponding estimators.
`AIC`	the Akaike information criterion.
`BIC`	the Bayesian information criterion.
`tau1, tau2`	values for tau1 and tau2, depending on the considered parameterization.
`pearson.res`	Pearson’s residuals.
`mod.pearson.res`	modified Pearson’s residuals.
`quant.res`	quantile residuals.
`convergence`	logical. If convergence was attained.
`dist`	BP (the beta prime distribution).
`param`	The specified parameterization.
`mu.x`	design matrix for mu.
`sigma.x`	design matrix for sigma.

Author(s)

Diego Gallardo and Marcelo Bourguignon.

References

Bourguignon, M., Gallardo, D.I. (2025) A general and unified parameterization of the beta distribution: A flexible and robust beta regression model. Statistica Neerlandica, 79(2), e70007.

Examples

```
set.seed(2100)
n=100; x1=rnorm(max(n)) ##drawing covariates, the same for mu and sigma
mu=exp(0.5-0.4*x1); sigma=exp(-0.1+0.05*x1)
y=rRBP(n, mu, sigma, param="MD") ## model parameterized in the median
data=list(y=y, x1=x1)
aux.RBP=fit.RBP(y~x1, sigma.formula=~x1, data=data, param="MD")
summary(aux.RBP)
qqnorm(res(aux.RBP, type="mod.pearson"))
```

`RGA` *Tools for a reparameterized gamma regression model*

Description

A set of functions related to the reparameterized gamma regression model based on different measures of central tendency: mean, median, mode, geometric mean or harmonic mean.

Usage

```
fit.RGA(formula = formula(data), sigma.formula=~1, data, param="AM")
dRGA(x, mu=1, sigma=1, param="AM", log=FALSE)
pRGA(q, mu=1, sigma=1, param="AM", lower.tail = TRUE, log.p = FALSE)
qRGA(p, mu=1, sigma=1, param="AM", lower.tail = TRUE, log.p = FALSE)
rRGA(n, mu=1, sigma=1, param="AM")
```

Arguments

`mu, sigma`	vector of parameter values
`formula`	an object of class "*formula*" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`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), typically the environment from which lm is called.
`sigma.formula`	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. sigma.formula=~x1+x2.
`param`	parameterization used for the model. "AM" for mean, "MD" for median, "MO" for mode, "GM" for geometric mean, and "HM" for harmonic mean.
`x, q`	vector of quantiles
`p`	vector of probabilities
`n`	number of observations. If length$(n) > 1$, the length is taken to be the number required.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE, probabilities are $P(X \leq x)$ otherwise, $P(X > x)$.

Details

The parameterization for the reparameterized beta distribution is given by

$$f(x;\mu,\sigma) = \frac{(\delta/\mu)^{\delta+\tau}}{\Gamma(\delta+\tau)} x^{\delta+\tau-1} \mathrm{e}^{-\delta x/\mu}, \quad y > 0,$$

where $\delta = \delta(\sigma) = (\sqrt{\sigma(\sigma+4)} + \sigma)/2$, $0 < \mu < 1$, $\sigma > 0$ and τ is a constant. The following cases are highlighted:

- param="AM": $\tau = 0$ and μ represents the mean of the distribution.

- param="GM": $\tau = 1/2$ and μ represents the geometric mean of the distribution.

- param="MD": $\tau = 1/3$ and μ represents the median of the distribution.

- param="MO" or ="HM": $\tau = 1$ and μ represents the mode or the harmonic mean of the distribution.

Suppose the central tendency and the concentration parameter of Y_i satisfies the following functional relations

$$\log(\mu_i) = \mathbf{x}_i^\top \boldsymbol{\xi} \quad \text{and} \quad \log(\sigma_i) = \eta_{2i} = \mathbf{z}_i^\top \boldsymbol{\nu},$$

where $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_p)^\top$ and $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_q)^\top$ are vectors of unknown regression coefficients which are assumed to be functionally independent, $\boldsymbol{\xi} \in \mathbb{R}^p$ and $\boldsymbol{\nu} \in \mathbb{R}^q$, with $p + q < n$, and $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\top$ and $\mathbf{z}_i = (z_{i1}, \ldots, z_{iq})^\top$ are observations on p and q known regressors, for $i = 1, \ldots, n$. Furthermore, we assume that the covariate matrices $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$ and $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)^\top$ have rank p and q, respectively.

For this model, the Pearson's residuals are given by

$$r_i = \frac{y_i - m_i}{s_i}, \quad i = 1, \ldots, n,$$

where

$$m_i = \mu_i \left(1 + \frac{\tau}{\xi(\sigma_i,\tau)}\right) \quad \text{and} \quad s_i = \frac{\mu}{\xi(\sigma_i,\tau)} \sqrt{\tau + \xi(\sigma_i,\tau)},$$

where $\xi(\sigma_i,\tau) = (\sqrt{\sigma_i(\sigma_i + 4\tau)} + \sigma_i)/2$. On the other hand, the modified Pearson's residuals are given by

$$r_i^* = \frac{\log(y_i) - m_i^*}{s_i^*}, \quad i = 1, \ldots, n,$$

where

$$m_i^* = \psi(\tau + \xi(\sigma_i,\tau)) + \log\mu_i - \log\xi(\sigma_i,\tau)) \quad \text{and} \quad s_i^* = \sqrt{\psi'(\tau + \xi(\sigma_i,\tau))},$$

with $\psi(\cdot)$ and $\psi'(\cdot)$ denoting the digamma and trigamma functions, respectively. Finally, the quantile residuals are given by

$$r_i^q = \Phi^{-1}\left(\frac{\gamma(\tau + \xi(\sigma_i,\tau), \xi(\sigma_i,\tau) y_i/\mu_i}{\Gamma(\tau + \xi(\sigma_i,\tau))}\right), \quad i = 1, \ldots, n,$$

where $\Phi^{-1}(\cdot)$ denotes the inverse of the cumulative distribution function for the standard normal model and $\gamma(a,z) = \int_0^z t^{a-1}\mathrm{e}^{-t}\mathrm{d}t$ is the lower incomplete gamma function and $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1}\mathrm{e}^{-\omega}\mathrm{d}\omega$ is the gamma function. dRGA gives the density, pRGA gives the distribution function, qRGA gives the quantile function, and rRGA generates random deviates from the beta distribution with the specified parameterization.

Value

an object of class "rregm" is returned. The object returned for this functions is a list containing the following components:

`estimate`	A matrix containing the estimates and standard errors.
`logLik`	the log-likelihood function evaluated at the corresponding estimators.
`AIC`	the Akaike information criterion.
`BIC`	the Bayesian information criterion.
`tau1, tau2`	values for tau1 and tau2, depending on the considered parameterization.
`pearson.res`	Pearson's residuals.
`mod.pearson.res`	modified Pearson's residuals.
`quant.res`	quantile residuals.
`convergence`	logical. If convergence was attained.
`dist`	GA (the gamma distribution).
`param`	The specified parameterization.
`mu.x`	design matrix for mu.
`sigma.x`	design matrix for sigma.

Author(s)

Diego Gallardo and Marcelo Bourguignon.

References

Bourguignon, M., Gallardo, D.I. (2025) A general and unified class of gamma regression models. Chemometrics and Intelligent Laboratory Systems, 261, 105382.

Examples

```
set.seed(2100)
n=100; x1=rnorm(max(n)) ##drawing covariates, the same for mu and sigma
mu=exp(0.5-0.4*x1); sigma=exp(-0.1+0.05*x1)
y=rRGA(n, mu, sigma, param="MD") ## model parameterized in the median
data=list(y=y, x1=x1)
aux.RGA=fit.RGA(y~x1, sigma.formula=~x1, data=data, param="MD")
summary(aux.RGA)
qqnorm(res(aux.RGA, type="mod.pearson"))
```

RIG *Tools for a reparameterized inverse gamma regression model*

Description

A set of functions related to the reparameterized inverse gamma regression model based on different measures of central tendency: mean, median, mode, geometric mean or harmonic mean.

Usage

```
IGAM(mu.link = "log", sigma.link = "log")
IGGM(mu.link = "log", sigma.link = "log")
IGHM(mu.link = "log", sigma.link = "log")
IGMD(mu.link = "log", sigma.link = "log")
IGMO(mu.link = "log", sigma.link = "logshiftto1")
dIGAM(x, mu = 1, sigma = 1, log = FALSE)
dIGGM(x, mu = 1, sigma = 1, log = FALSE)
dIGHM(x, mu = 1, sigma = 1, log = FALSE)
dIGMD(x, mu = 1, sigma = 1, log = FALSE)
dIGMO(x, mu = 1, sigma = 1.5, log = FALSE)
dRIG(x, mu=1, sigma=1.5, param="AM", log=FALSE)
fit.RIG(formula = formula(data), sigma.formula=~1, data, param="AM")
pIGAM(q, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pIGGM(q, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pIGHM(q, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pIGMD(q, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
pIGMO(q, mu = 1, sigma = 1.5, lower.tail = TRUE, log.p = FALSE)
pRIG(q, mu=1, sigma=1.5, param="AM", lower.tail = TRUE, log.p = FALSE)
qIGAM(p, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qIGGM(p, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qIGHM(p, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qIGMD(p, mu = 1, sigma = 1, lower.tail = TRUE, log.p = FALSE)
qIGMO(p, mu = 1, sigma = 1.5, lower.tail = TRUE, log.p = FALSE)
qRIG(p, mu=1, sigma=1.5, param="AM", lower.tail = TRUE, log.p = FALSE)
rIGAM(n, mu = 1, sigma = 1)
rIGGM(n, mu = 1, sigma = 1)
rIGHM(n, mu = 1, sigma = 1)
rIGMD(n, mu = 1, sigma = 1)
rIGMO(n, mu = 1, sigma = 1.5)
rRIG(n, mu=1, sigma=1.5, param="AM")
```

Arguments

`mu.link`	the mu link function with default log
`sigma.link`	the sigma link function with default log
`mu, sigma`	vector of parameter values

`formula`	an object of class "*formula*" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`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), typically the environment from which lm is called.
`sigma.formula`	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. sigma.formula=~x1+x2.
`param`	parameterization used for the model. "AM" for mean, "MD" for median, "MO" for mode, "GM" for geometric mean, and "HM" for harmonic mean.
`x, q`	vector of quantiles
`p`	vector of probabilities
`n`	number of observations. If length$(n) > 1$, the length is taken to be the number required.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE, probabilities are $P(X \le x)$ otherwise, $P(X > x)$.

Details

The parameterization for the reparameterized inverse gamma distribution is given by

$$f(y;\mu,\sigma) = \frac{(\sigma\mu)^{\sigma+\tau}}{\Gamma(\sigma+\tau)} y^{-\sigma-\tau-1} \mathrm{e}^{-\frac{\sigma\mu}{y}}, \quad y > 0,$$

where $0 < \mu < 1$, $\sigma > 0$ and τ is a constant. The following cases are highlighted:

- param="AM": $\tau = 1$ and μ represents the mean of the distribution.

- param="GM": $\tau = 1/2$ and μ represents the geometric mean of the distribution.

- param="HM": $\tau = 0$ and μ represents the harmonic mean of the distribution.

- param="MO": $\tau = -1$ and μ represents the mode of the distribution.

- param="MD": $\tau = 1/2$ and μ represents the median of the distribution.

Suppose the central tendency and the concentration parameter of Y_i satisfies the following functional relations

$$\log(\mu_i) = \mathbf{x}_i^\top \boldsymbol{\xi} \quad \text{and} \quad \log(\sigma_i) = \eta_{2i} = \mathbf{z}_i^\top \boldsymbol{\nu},$$

$\boldsymbol{\xi} = (\xi_1, \ldots, \xi_p)^\top$ and $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_q)^\top$ are vectors of unknown regression coefficients which are assumed to be functionally independent, $\boldsymbol{\xi} \in \mathbb{R}^p$ and $\boldsymbol{\nu} \in \mathbb{R}^q$, with $p + q < n$, and $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\top$ and $\mathbf{z}_i = (z_{i1}, \ldots, z_{iq})^\top$ are observations on p and q known regressors, for $i = 1, \ldots, n$. Furthermore, we assume that the covariate matrices $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$ and $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)^\top$ have rank p and q, respectively.

For this model, the Pearson's residuals are given by

$$r_i = \frac{y_i - m_i}{s_i}, \quad i = 1, \ldots, n,$$

where

$$m_i = \frac{\mu_i \sigma_i}{\sigma_i + \tau_2} \quad \text{and} \quad s_i = \frac{\mu_i \sigma_i}{(\sigma_i + \tau - 1)}(\sigma_i + \tau - 2)^{-1/2}.$$

Note that the Pearson's residuals are well defined as long as $\sigma_i + \tau > 2, \forall i = 1, \ldots, n.$ On the other hand, the modified Pearson's residuals are given by

$$r_i^* = \frac{-\log(y_i) - m_i^*}{s_i^*}, \quad i = 1, \ldots, n,$$

where

$$m_i^* = \psi(\sigma_i + \tau) - \log(\mu_i \sigma_i) \quad \text{and} \quad s_i^* = \sqrt{\psi'(\sigma_i + \tau)},$$

with $\psi(\cdot)$ and $\psi'(\cdot)$ denoting the digamma and trigamma functions, respectively. Finally, the quantile residuals are given by

$$r_i^q = \Phi^{-1}\left(\frac{\gamma(\sigma_i + \tau, \mu_i \sigma_i / y_i)}{\Gamma(\sigma_i + \tau)}\right), \quad i = 1, \ldots, n,$$

where $\Phi^{-1}(\cdot)$ denotes the inverse of the cumulative distribution function for the standard normal model and $\gamma(a, z) = \int_0^z t^{a-1} \mathrm{e}^{-t} \mathrm{d}t$ is the lower incomplete gamma function and $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1} \mathrm{e}^{-\omega} \mathrm{d}\omega$ is the gamma function. dRIG gives the density, pRIG gives the distribution function, qRIG gives the quantile function, and rRIG generates random deviates from the inverse gamma distribution with the specified parameterization. In addition, dIGXX, pIGXX, qIGXX and rIGXX also provides the equivalent functions for a specified parameterization for XX: AM (mean), GM (geometric mean), HM (harmonic mean), MD (median) and MO (mode). For instance, dIGAM gives the density for the inverse gamma model parameterized in the mean, pIGGM gives the distribution function for the inverse gamma model parameterized in the geometric mean and so on. Finally, the functions IGAM, IGGM, IGHM, IGMD and IGMO also provide a framework to fit models with *gamlss*.

Value

an object of class "rregm" is returned. The object returned for this functions is a list containing the following components:

`estimate`	A matrix containing the estimates and standard errors.
`logLik`	the log-likelihood function evaluated at the corresponding estimators.
`AIC`	the Akaike information criterion.
`BIC`	the Bayesian information criterion.
`tau1, tau2`	values for tau1 and tau2, depending on the considered parameterization.
`pearson.res`	Pearson's residuals.
`mod.pearson.res`	modified Pearson's residuals.
`quant.res`	quantile residuals.
`convergence`	logical. If convergence was attained.
`dist`	IG (the inverse gamma distribution).
`param`	The specified parameterization.
`mu.x`	design matrix for mu.
`sigma.x`	design matrix for sigma.

Author(s)

Diego Gallardo and Marcelo Bourguignon.

References

Bourguignon, M., Gallardo, D.I. (2025) A general and unified class of gamma regression models. Chemometrics and Intelligent Laboratory Systems, 261, 105382.

Examples

```
set.seed(2100)
n=100; x1=rnorm(max(n)) ##drawing covariates, the same for mu and sigma
mu=exp(0.5-0.4*x1); sigma=exp(-0.1+0.05*x1)
y=rRIG(n, mu, sigma, param="MD") ## model parameterized in the median
data=list(y=y, x1=x1)
aux.RIG=fit.RIG(y~x1, sigma.formula=~x1, data=data, param="MD")
summary(aux.RIG)
qqnorm(res(aux.RIG, type="mod.pearson"))
#The inverse gamma model parameterized in the median also can be fitted using gamlss
#gamlss(y~x1, sigma.formula=~x1, data=data, family=IGMD)
```

`tools.rregm` *Print a summary for a object of the "rregm" class.*

Description

Tools for a objects of the "rregm" class.

Usage

```
res(object, type="pearson")
## S3 method for class 'rregm'
AIC(object, ..., k=2)
## S3 method for class 'rregm'
BIC(object, ...)
## S3 method for class 'rregm'
coef(object, ...)
## S3 method for class 'rregm'
logLik(object, ...)
## S3 method for class 'rregm'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
## S3 method for class 'rregm'
summary(object, ...)
```

Arguments

`x, object`	an object of the "rregm" class.
`type`	type of residuals to be presented: pearson (default), mod.pearson or quantile.
`digits`	minimal number of significant digits
`k`	numeric, the penalty per parameter to be used; the default k = 2 is the classical AIC.
`...`	for extra arguments.

Details

Supported regression models are: - reparametrized beta - reparametrized gamma - reparametrized beta prime - reparametrized inverse gamma

Value

A complete summary for the coefficients extracted from a "rregm" object.

Author(s)

Diego Gallardo and Marcelo Bourguignon.

References

Gallardo and Bourguignon (2022).

Examples

```
set.seed(2100)
n=100; x1=rnorm(max(n)) ##drawing covariates, the same for mu and sigma
mu=exp(0.5-0.4*x1); sigma=exp(-0.1+0.05*x1)
y=rRGA(n, mu, sigma, param="MD") ## model parameterized in the median
data=list(y=y, x1=x1)
aux.RGA=fit.RGA(y~x1, sigma.formula=~x1, data=data, param="MD")
summary(aux.RGA)
qqnorm(res(aux.RGA, type="mod.pearson"))
```

Index