

# Package ‘rregm’

April 11, 2026

**Type** Package

**Title** Reparameterized Regression Models

**Version** 1.1

**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](https://doi.org/10.1016/j.chemolab.2025.105382)> and Bourguignon and Gallardo (2025b) <[doi:10.1111/stan.70007](https://doi.org/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** 2026-04-10 23:50:02 UTC

## Contents

RBE . . . . .	2
RBP . . . . .	5
RGA . . . . .	8
RIG . . . . .	11
RLN . . . . .	14
tools.rregm . . . . .	17

<b>Index</b>	<b>19</b>
--------------	-----------

**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**

<code>mu.link</code>	the mu link function with default logit
<code>sigma.link</code>	the sigma link function with default log
<code>mu, sigma</code>	vector of parameter values

formula	an object of class " <i>formula</i> " (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 <i>as.data.frame</i> 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 <i>lm</i> is called.
sigma.formula	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. <code>sigma.formula=~x1+x2</code> .
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 <code>length(n) &gt; 1</code> , the length is taken to be the number required.
log, log.p	logical; if TRUE, probabilities p are given as <code>log(p)</code> .
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  $\tau_1$  and  $\tau_2$  are constant. The following cases are highlighted:

- param="AM":  $\tau_1 = \tau_2 = 0$  and  $\mu$  represents the mean of the distribution.
- param="GM":  $\tau_1 = \tau_2 = 1/2$  and  $\mu$  represents the geometric mean of the distribution.
- param="HM":  $\tau_1 = \tau_2 = 1$  and  $\mu$  represents the harmonic mean of the distribution.
- param="MO":  $\tau_1 = 1$  and  $\tau_2 = 2$  and  $\mu$  represents the mode of the distribution.
- param="MD":  $\tau_1 = 1/2$  and  $\tau_2 = 0$  and  $\mu$  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,  $\boldsymbol{\xi} = (\xi_1, \dots, \xi_p)^\top$  and  $\boldsymbol{\nu} = (\nu_1, \dots, \nu_q)^\top$  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  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^\top$  and  $\mathbf{z}_i = (z_{i1}, \dots, z_{iq})^\top$  are observations on  $p$  and  $q$  known regressors, for  $i = 1, \dots, n$ . Furthermore, we assume that the covariate matrices  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$  and  $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)^\top$  have rank  $p$  and  $q$ , respectively.

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

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

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

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

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

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

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,  $B_x(\alpha, \beta) = \int_0^x \omega^{\alpha-1} (1 - \omega)^{\beta-1} d\omega$  is the incomplete beta function,  $B(\alpha, \beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha + \beta)$  is the beta function and  $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1} e^{-\omega} d\omega$  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, dBEM 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 " <i>formula</i> " (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 <i>as.data.frame</i> 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  $\tau$  is a constant. The following cases are highlighted:

- param="AM":  $\tau = 0$  and  $\mu$  represents the mean of the distribution.
- param="GM":  $\tau = 1/2$  and  $\mu$  represents the geometric mean of the distribution.
- param="MD":  $\tau = 1/3$  and  $\mu$  represents the median of the distribution.
- param="MO" or "HM":  $\tau = 1$  and  $\mu$  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

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

where  $\boldsymbol{\xi} = (\xi_1, \dots, \xi_p)^\top$  and  $\boldsymbol{\nu} = (\nu_1, \dots, \nu_q)^\top$  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  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^\top$  and  $\mathbf{z}_i = (z_{i1}, \dots, z_{iq})^\top$  are observations on  $p$  and  $q$  known regressors, for  $i = 1, \dots, n$ . Furthermore, we assume that the covariate matrices  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$  and  $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)^\top$  have rank  $p$  and  $q$ , respectively.

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

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

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  $\sigma_i - \tau_2 > 1$ ,  $\forall i = 1, \dots, n$ . On the other hand, the modified Pearson's residuals are given by

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

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,  $B_x(\alpha, \beta) = \int_0^x \omega^{\alpha-1}(1-\omega)^{\beta-1}d\omega$  is the incomplete beta function,  $B(\alpha, \beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha + \beta)$  is the beta function and  $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1}e^{-\omega}d\omega$  is the gamma function. `drBP` gives the density, `prBP` gives the distribution function, `qrBP` gives the quantile function, and `rBP` 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:

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

<code>mu, sigma</code>	vector of parameter values
<code>formula</code>	an object of class " <i>formula</i> " (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'.
<code>data</code>	an optional data frame, list or environment (or object coercible by <i>as.data.frame</i> 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 <i>lm</i> is called.
<code>sigma.formula</code>	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. <code>sigma.formula=~x1+x2</code> .
<code>param</code>	parameterization used for the model. "AM" for mean, "MD" for median, "MO" for mode, "GM" for geometric mean, and "HM" for harmonic mean.
<code>x, q</code>	vector of quantiles
<code>p</code>	vector of probabilities
<code>n</code>	number of observations. If $\text{length}(n) > 1$ , the length is taken to be the number required.
<code>log, log.p</code>	logical; if TRUE, probabilities p are given as $\log(p)$ .
<code>lower.tail</code>	logical; if TRUE, probabilities are $P(X \leq x)$ otherwise, $P(X > x)$ .

### Details

The parameterization for the reparameterized gamma distribution is given by

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

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

- param="AM":  $\tau = 0$  and  $\mu$  represents the mean of the distribution.
- param="GM":  $\tau = 1/2$  and  $\mu$  represents the geometric mean of the distribution.
- param="MD":  $\tau = 1/3$  and  $\mu$  represents the median of the distribution.
- param="MO" or "HM":  $\tau = 1$  and  $\mu$  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  $\boldsymbol{\xi} = (\xi_1, \dots, \xi_p)^\top$  and  $\boldsymbol{\nu} = (\nu_1, \dots, \nu_q)^\top$  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  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^\top$  and  $\mathbf{z}_i = (z_{i1}, \dots, z_{iq})^\top$  are observations on  $p$  and  $q$  known regressors, for  $i = 1, \dots, n$ . Furthermore, we assume that the covariate matrices  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$  and  $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)^\top$  have rank  $p$  and  $q$ , respectively.

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

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

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

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

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

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

where  $\Phi^{-1}(\cdot)$  denotes the inverse of the cumulative distribution function for the standard normal model and  $\gamma(a, z) = \int_0^z t^{a-1} e^{-t} dt$  is the lower incomplete gamma function and  $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1} e^{-\omega} d\omega$  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 gamma 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"))
```

**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 " <i>formula</i> " (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 <i>as.data.frame</i> 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 <i>lm</i> is called.
sigma.formula	a formula object for fitting a model to the sigma parameter, as in the formula above, e.g. <code>sigma.formula=~x1+x2</code> .
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 <code>length(n) &gt; 1</code> , the length is taken to be the number required.
log, log.p	logical; if TRUE, probabilities p are given as <code>log(p)</code> .
lower.tail	logical; if TRUE, probabilities are $P(X \leq x)$ otherwise, $P(X > x)$ .

## Details

The parameterization for the reparameterized inverse gamma distribution is given by

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

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

- param="AM":  $\tau = 1$  and  $\mu$  represents the mean of the distribution.
- param="GM":  $\tau = 1/2$  and  $\mu$  represents the geometric mean of the distribution.
- param="HM":  $\tau = 0$  and  $\mu$  represents the harmonic mean of the distribution.
- param="MO":  $\tau = -1$  and  $\mu$  represents the mode of the distribution.
- param="MD":  $\tau = 1/2$  and  $\mu$  represents the median of the distribution.

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

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

$\boldsymbol{\xi} = (\xi_1, \dots, \xi_p)^\top$  and  $\boldsymbol{\nu} = (\nu_1, \dots, \nu_q)^\top$  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  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^\top$  and  $\mathbf{z}_i = (z_{i1}, \dots, z_{iq})^\top$  are observations on  $p$  and  $q$  known regressors, for  $i = 1, \dots, n$ . Furthermore, we assume that the covariate matrices  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$  and  $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)^\top$  have rank  $p$  and  $q$ , respectively.

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

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

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  $\sigma_i + \tau > 2$ ,  $\forall i = 1, \dots, n$ . On the other hand, the modified Pearson's residuals are given by

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

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

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

where  $\Phi^{-1}(\cdot)$  denotes the inverse of the cumulative distribution function for the standard normal model and  $\gamma(a, z) = \int_0^z t^{a-1} e^{-t} dt$  is the lower incomplete gamma function and  $\Gamma(\alpha) = \int_0^\infty \omega^{\alpha-1} e^{-\omega} d\omega$  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)
```

---

 RLN

---

*Tools for a reparameterized log-normal regression model*


---

**Description**

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

**Usage**

```
fit.RLN(formula = formula(data), sigma.formula=~1, data, param="AM")
dRLN(x, mu=1, sigma=1, param="AM", log=FALSE)
pRLN(q, mu=1, sigma=1, param="AM", lower.tail = TRUE, log.p = FALSE)
qRLN(p, mu=1, sigma=1, param="AM", lower.tail = TRUE, log.p = FALSE)
rRLN(n, mu=1, sigma=1, param="AM")
```

**Arguments**

mu, sigma	vector of parameter values
formula	an object of class " <i>formula</i> " (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 <i>as.data.frame</i> 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 <i>lm</i> is called.

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

### Details

The parameterization for the reparameterized log-normal distribution is given by

$$f(x; \mu, \sigma) = \frac{1}{x \phi \sqrt{2\pi}} \exp\left[-\frac{(\ln x - \theta)^2}{2\phi^2}\right], \quad y > 0,$$

where  $\theta = \log(\mu) - \tau\phi$ ,  $\phi = \sqrt{\sigma}$ ,  $\mu > 0$ ,  $\sigma > 0$  and  $\tau$  is a constant. The following cases are highlighted:

- param="MO":  $\tau = -1$  and  $\mu$  represents the mode or the harmonic mean of the distribution.
- param="HM":  $\tau = -1/2$  and  $\mu$  represents the harmonic mean of the distribution.
- param="MD" or "GM":  $\tau = 0$  and  $\mu$  represents the median or the geometric mean of the distribution.
- param="AM":  $\tau = 1/2$  and  $\mu$  represents the 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  $\boldsymbol{\xi} = (\xi_1, \dots, \xi_p)^\top$  and  $\boldsymbol{\nu} = (\nu_1, \dots, \nu_q)^\top$  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  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^\top$  and  $\mathbf{z}_i = (z_{i1}, \dots, z_{iq})^\top$  are observations on  $p$  and  $q$  known regressors, for  $i = 1, \dots, n$ . Furthermore, we assume that the covariate matrices  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^\top$  and  $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_n)^\top$  have rank  $p$  and  $q$ , respectively.

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

$$r_i = \frac{y_i - \exp(\theta_i + \phi_i^2/2)}{\sqrt{((\exp(\phi_i^2) - 1) \exp(2 * \theta_i + \phi_i^2))}}, \quad i = 1, \dots, n.$$

On the other hand, the modified Pearson's residuals are given by

$$r_i^* = \frac{\log(y_i) - \theta_i}{\phi_i}, \quad i = 1, \dots, n,$$

which coincide with the quantile residuals. dRLN gives the density, pRLN gives the distribution function, qRLN gives the quantile function, and rRLN generates random deviates from the log-normal 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=rRLN(n, mu, sigma, param="MD") ## model parameterized in the median
data=list(y=y, x1=x1)
aux.RLN=fit.RLN(y~x1, sigma.formula=~x1, data=data, param="MD")
summary(aux.RLN)
qqnorm(res(aux.RLN, type="mod.pearson"))
```

---

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

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"))
```

# Index

AIC.rregm (tools.rregm), 17

BEAM (RBE), 2  
BEGM (RBE), 2  
BEHM (RBE), 2  
BEMD (RBE), 2  
BEMO (RBE), 2  
BIC.rregm (tools.rregm), 17

coef.rregm (tools.rregm), 17

dBEAM (RBE), 2  
dBEGM (RBE), 2  
dBEHM (RBE), 2  
dBEMD (RBE), 2  
dBEMO (RBE), 2  
dIGAM (RIG), 11  
dIGGM (RIG), 11  
dIGHM (RIG), 11  
dIGMD (RIG), 11  
dIGMO (RIG), 11  
dRBE (RBE), 2  
dRBP (RBP), 5  
dRGA (RGA), 8  
dRIG (RIG), 11  
dRLN (RLN), 14

fit.RBE (RBE), 2  
fit.RBP (RBP), 5  
fit.RGA (RGA), 8  
fit.RIG (RIG), 11  
fit.RLN (RLN), 14

IGAM (RIG), 11  
IGGM (RIG), 11  
IGHM (RIG), 11  
IGMD (RIG), 11  
IGMO (RIG), 11

logLik.rregm (tools.rregm), 17

pBEAM (RBE), 2  
pBEGM (RBE), 2  
pBEHM (RBE), 2  
pBEMD (RBE), 2  
pBEMO (RBE), 2  
pIGAM (RIG), 11  
pIGGM (RIG), 11  
pIGHM (RIG), 11  
pIGMD (RIG), 11  
pIGMO (RIG), 11  
pRBE (RBE), 2  
pRBP (RBP), 5  
pRGA (RGA), 8  
pRIG (RIG), 11  
print.rregm (tools.rregm), 17  
pRLN (RLN), 14

qBEAM (RBE), 2  
qBEGM (RBE), 2  
qBEHM (RBE), 2  
qBEMD (RBE), 2  
qBEMO (RBE), 2  
qIGAM (RIG), 11  
qIGGM (RIG), 11  
qIGHM (RIG), 11  
qIGMD (RIG), 11  
qIGMO (RIG), 11  
qRBE (RBE), 2  
qRBP (RBP), 5  
qRGA (RGA), 8  
qRIG (RIG), 11  
qRLN (RLN), 14

RBE, 2  
rBEAM (RBE), 2  
rBEGM (RBE), 2  
rBEHM (RBE), 2  
rBEMD (RBE), 2  
rBEMO (RBE), 2  
RBP, 5

`res (tools.rregm)`, 17  
`RGA`, 8  
`RIG`, 11  
`rIGAM (RIG)`, 11  
`rIGGM (RIG)`, 11  
`rIGHM (RIG)`, 11  
`rIGMD (RIG)`, 11  
`rIGMO (RIG)`, 11  
`RLN`, 14  
`rRBE (RBE)`, 2  
`rRBP (RBP)`, 5  
`rRGA (RGA)`, 8  
`rRIG (RIG)`, 11  
`rRLN (RLN)`, 14  
  
`summary.rregm (tools.rregm)`, 17  
  
`tools.rregm`, 17