Package 'BayesLN'

May 2, 2022

Title Bayesian Inference for Log-Normal Data

Version 0.2.6

Description Bayesian inference under log-normality assumption must be performed very carefully. In fact, under the common priors for the variance, useful quantities in the original data scale (like mean and quantiles) do not have posterior moments that are finite (Fabrizi et al. 2012 <doi:10.1214/12-BA733>). This package allows to easily carry out a proper Bayesian inferential procedure by fixing a suitable distribution (the generalized inverse Gaussian) as prior for the variance. Functions to estimate several kind of means (unconditional, conditional and conditional under a mixed model) and quantiles (unconditional and conditional) are provided.

Depends R (>= 3.5.0)

Imports optimx, GeneralizedHyperbolic, gsl, coda, Rcpp (>= 0.12.17), MASS, lme4, data.table, Matrix, methods

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

LinkingTo Rcpp, RcppEigen

Date 2022-05-02

NeedsCompilation yes

Author Aldo Gardini [aut, cre] (<https://orcid.org/0000-0002-2164-5815>),
    Enrico Fabrizi [aut] (<https://orcid.org/0000-0003-2504-7043>),
    Carlo Trivisano [aut] (<https://orcid.org/0000-0002-5991-4902>)

Maintainer Aldo Gardini <aldo.gardini2@unibo.it>

Repository CRAN

Date/Publication 2022-05-02 11:30:18 UTC

R topics documented:

EPA09 *Chrysene concentration data*

Description

Vector of 8 observations of chrysene concentration (ppb) found in water samples.

Usage

EPA09

Format

Numeric vector.

Source

USEPA. *Statistical analysis of groundwater monitoring data at rcra facilities: Unifed guidance.* Technical report, Office of Resource Conservation and Recovery, Program Implementation and Information Division, U.S. Environmental Protection Agency, Washington, D.C. (2009).

---

fatigue	*Low cycle fatigue data*

Description

Data frame of 22 observations in 2 variables

Usage

```
fatigue
```

Format

Dataframe with variables:

stress: stress factor.

cycle: number of test cycles.

Source

Upadhyay, S. K., and M. Peshwani. *Posterior analysis of lognormal regression models using the Gibbs sampler.* Statistical Papers 49.1 (2008): 59-85.

GH_MGF	*GH Moment Generating Function*

Description

Function that implements the moment generating function of the Generalized Hyperbolyc (GH) distribution.

Usage

```
GH_MGF(r, mu = 0, delta, alpha, lambda, beta = 0)
```

Arguments

r	Coefficient of the MGF. Can be viewd also as the order of the log-GH moments.
mu	Location parameter, default set to 0.
delta	Concentration parameter, must be positive.
alpha	Tail parameter, must be positive and greater than the modulus of beta.
lambda	Shape parameter.
beta	Skewness parameter, default set to 0 (symmetric case).

Details

This function allows to evaluate the moment generating function of the GH distribution in the point r. It is defined only for points that are lower than the value of γ, that is defined as: $\gamma^2 = \alpha^2 - \beta^2$. For integer values of r, it could also be considered as the *r*-th raw moment of the log-GH distribution.

---

laminators   *Laminators*

Description

Data frame of 39 observations in 2 variables.

Usage

laminators

Format

Dataframe with variables:

Worker: label of the measured worker.

log_Y: logarithm of the measured Styrene concentration.

Source

R. H. Lyles, L. L. Kupper, and S. M. Rappaport. *Assessing regulatory compliance of occupational exposures via the balanced one-way random effects ANOVA model* Journal of Agricultural, Biological, and Environmental Statistics (1997).

LN_hierarchical   *Bayesian estimation of a log - normal hierarchical model*

Description

Function that estimates a log-normal linear mixed model with GIG priors on the variance components, in order to assure the existence of the posterior moments of key functionals in the original data scale like conditioned means or the posterior predictive distribution.

Usage

```
LN_hierarchical(
  formula_lme,
  data_lme,
  y_transf = TRUE,
  functional = c("Subject", "Marginal", "PostPredictive"),
  data_pred = NULL,
  order_moment = 2,
  nsamp = 10000,
  par_tau = NULL,
  par_sigma = NULL,
  var_pri_beta = 10000,
  inits = list(NULL),
  verbose = TRUE,
  burnin = 0.1 * nsamp,
  n_thin = 1
)
```

Arguments

formula_lme	A two-sided linear formula object describing both the fixed-effects and random-effects part of the model is required. For details see `lmer`.
data_lme	Optional data frame containing the variables named in `formula_lme`.
y_transf	Logical. If `TRUE`, the response variable is assumed already as log-transformed.
functional	Functionals of interest: `"Subject"` for subject-specific conditional mean, `"Marginal"` for the overall expectation and `"PostPredictive"` for the posterior predictive distribution.
data_pred	Data frame with the covariate patterns of interest for prediction. All the covariates present in the `data_lme` object must be included. If `NULL` the design matrix of the model is used.
order_moment	Order of the posterior moments that are required to be finite.
nsamp	Number of Monte Carlo iterations.
par_tau	List of vectors defining the triplets of hyperparaemters for each random effect variance (as many vectors as the number of specified random effects variances).
par_sigma	Vector containing the tiplet of hyperparameters for the prior of the data variance.
var_pri_beta	Prior variance for the model coefficients.
inits	List of object for initializing the chains. Objects with compatible dimensions must be named with `beta`, `sigma2` and `tau2`.
verbose	Logical. If `FALSE`, the messages from the Gibbs sampler are not shown.
burnin	Number of iterations to consider as burn-in.
n_thin	Number of thinning observations.

Details

The function allows to estimate a log-normal linear mixed model through a Gibbs sampler. The model equation is specified as in lmer model and the target functionals to estimate need to be declared. A weakly informative prior setting is automatically assumed, always keeping the finiteness of the posterior moments of the target functionals.

Value

The output list provided is composed of three parts. The object $par_prior contains the parameters fixed for the variance components priors. The object $samples contains the posterior samples for all the paramters. They are returned as a mcmc object and they can be analysed trough the functions contained in the coda package in order to check for the convergence of the algorithm. Finally, in $summaries an overview of the posteriors of the model parameters and of the target functionals is provided.

Examples

```
library(BayesLN)
# Load the dataset included in the package
data("laminators")
data_pred_new <- data.frame(Worker = unique(laminators$Worker))
Mod_est<-LN_hierarchical(formula_lme = log_Y~(1|Worker),
                         data_lme = laminators,
                         data_pred = data_pred_new,
                         functional = c("Subject","Marginal"),
                         order_moment = 2, nsamp = 50000, burnin = 10000)
```

LN_hier_existence   *Numerical evaluation of the log-normal conditioned means posterior*
                           moments

Description

Function that evaluates the existence conditions for moments of useful quantities in the original data scale when a log-normal linear mixed model is estimated.

Usage

```
LN_hier_existence(X, Z, Xtilde, order_moment = 2, s = 1, m = NULL)
```

Arguments

X	Design matrix for fixed effects.
Z	Design matrix for random effects.
Xtilde	Covariate patterns used for the leverage computation.

order_moment	Order of the posterior moments required to be finite.
s	Number of variances of the random effects.
m	Vector of size s (if s>1) that indicates the dimensions of the random effect vectors.

Details

This function computes the existence conditions for the moments up to order fixed by order_moment of the log-normal linear mixed model specified by the design matrices X and Z. It considers the prediction based on multiple covariate patterns stored in the rows of the Xtilde matrix.

Value

Both the values of the factors determining the existence condition and the values of the gamma parameters for the different variance components are provided.

LN_Mean	*Bayesian Estimate of the Log-normal Mean*

Description

This function produces a Bayesian estimate of the log-normal mean, assuming a GIG prior for the variance and an improper flat prior for the mean in the log scale.

Usage

```
LN_Mean(
  x,
  method = "weak_inf",
  x_transf = TRUE,
  CI = TRUE,
  alpha_CI = 0.05,
  type_CI = "two-sided",
  nrep = 1e+05
)
```

Arguments

x	Vector containing the sample.
method	String that indicates the prior setting to adopt. Choosing "weak_inf" a weakly informative prior setting is adopted, whereas selecting "optimal" the hyperparameters are aimed at minimizing the frequentist MSE.
x_transf	Logical. If TRUE, the x vector is assumed already log-transformed.
CI	Logical. With the default choice TRUE, the posterior credibility interval is computed.

alpha_CI   Level of alpha that determines the credibility (1-alpha_CI) of the posterior interval.

type_CI   String that indicates the type of interval to compute: `"two-sided"` (default), `"UCL"` (i.e. Upper Credible Limit) for upper one-sided intervals or `"LCL"` (i.e. Lower Credible Limit) for lower one-sided intervals.

nrep   Number of simulations for the computation of the credible intervals.

Details

Summarizing the posterior mean of the log-normal expectation might be delicate since several common priors used for the variance do not produces posteriors with finite moments. The proposal by Fabrizi and Trivisano (2012) of adopting a generalized inverse Gaussian (GIG) prior for the variance in the log scale σ^2 has been implemented. Moreover, they discussed how to specify the hyperparameters according to two different aims.

Firstly, a weakly informative prior allowed to produce posterior credible intervals with good frequentist properties, whereas a prior aimed at minimizing the point estimator MSE was proposed too. The choice between the two priors can be made through the argument method.

The point estimates are exact values, whereas the credible intervals are provided through simulations from the posterior distribution.

Value

The function returns a list which includes the prior and posterior parameters, the point estimate of the log-normal mean that consists in the mean of the posterior distribution of the functional $\exp\{\mu + \sigma^2/2\}$ and the posterior variance.

Source

Fabrizi, E., & Trivisano, C. *Bayesian estimation of log-normal means with finite quadratic expected loss*. Bayesian Analysis, 7(4), 975-996. (2012).

Examples

```
# Load data
data("NCBC")
# Optimal point estimator
LN_Mean(x = NCBC$al, x_transf = FALSE, method = "optimal", CI = FALSE)
# Weakly informative prior and interval estimation
LN_Mean(x = NCBC$al, x_transf = FALSE, type_CI = "UCL")
```

LN_MeanReg   *Bayesian Estimate of the conditional Log-normal Mean*

Description

This function produces a bayesian estimate of the conditional log-normal mean assuming a GIG prior for the variance and an improper prior for the regression coefficients of the linear regression in the log scale.

Usage

```
LN_MeanReg(
  y,
  X,
  Xtilde,
  method = "weak_inf",
  y_transf = TRUE,
  h = NULL,
  CI = TRUE,
  alpha_CI = 0.05,
  type_CI = "two-sided",
  nrep = 1e+05
)
```

Arguments

y	Vector of observations of the response variable.
X	Design matrix.
Xtilde	Matrix of covariate patterns for which an estimate is required.
method	String that indicates the prior setting to adopt. Choosing `"weak_inf"` a weakly informative prior setting is adopted, whereas selecting `"optimal"` the hyperparameters are aimed at minimizing the frequentist MSE.
y_transf	Logical. If `TRUE`, the y vector is already assumed as log-transformed.
h	Leverage. With the default option `NULL`, the average leverage is used.
CI	Logical. With the default choice `TRUE`, the posterior credibility interval is computed.
alpha_CI	Level of alpha that determines the credibility (1-`alpha_CI`) of the posterior interval.
type_CI	String that indicates the type of interval to compute: `"two-sided"` (default), `"UCL"` (i.e. Upper Credible Limit) for upper one-sided intervals or `"LCL"` (i.e. Lower Credible Limit) for lower one-sided intervals.
nrep	Number of simulations.

Details

In this function the same procedure as LN_Mean is implemented allowing for the inclusion of covariates. Bayesian point and interval estimates for the response variabile in the original scale are provided considering the model: $log(y_i) = X\beta$.

Value

The function returns a list including the prior and posterior parameters, the point estimate of the log-normal mean conditioned with respect to the covariate points included in Xtilde. It consists of the mean of the posterior distribution for the functional $\exp\{\tilde{x}_i^T\beta + \sigma^2/2\}$ and the posterior variance.

Source

Fabrizi, E., & Trivisano, C. *Bayesian Conditional Mean Estimation in Log-Normal Linear Regression Models with Finite Quadratic Expected Loss.* Scandinavian Journal of Statistics, 43(4), 1064-1077. (2016).

Examples

```
library(BayesLN)
data("fatigue")

# Design matrices
Xtot <- cbind(1, log(fatigue$stress), log(fatigue$stress)^2)
X <- Xtot[-c(1,13,22),]
y <- fatigue$cycle[-c(1,13,22)]
Xtilde <- Xtot[c(1,13,22),]
#Estimation
LN_MeanReg(y = y,
           X = X, Xtilde = Xtilde,
           method = "weak_inf", y_transf = FALSE)
```

LN_Quant *Bayesian estimate of the log-normal quantiles*

Description

This function produces an estimate for the log-normal distribution quantile of fixed level quant.

Usage

```
LN_Quant(
  x,
  quant,
  method = "weak_inf",
```

```
    x_transf = TRUE,
    guess_s2 = NULL,
    CI = TRUE,
    alpha_CI = 0.05,
    type_CI = "two-sided",
    method_CI = "exact",
    rel_tol_CI = 1e-05,
    nrep_CI = 1e+06
)
```

Arguments

x	Vector of data used to estimate the quantile.
quant	Number between 0 and 1 that indicates the quantile of interest.
method	String that indicates the prior setting to adopt. Choosing `"weak_inf"` a weakly informative prior setting is adopted, whereas selecting `"optimal"` the hyperparameters are fixed trough a numerical optimization algorithm aimed at minimizing the frequentist MSE.
x_transf	Logical. If `TRUE`, the x vector is assumed already log-transformed.
guess_s2	Specification of a guess for the variance if available. If not, the sample estimate is used.
CI	Logical. With the default choice `TRUE`, the posterior credibility interval is computed.
alpha_CI	Level of alpha that determines the credibility (1-alpha_CI) of the posterior interval.
type_CI	String that indicates the type of interval to compute: `"two-sided"` (default), `"UCL"` (i.e. Upper Credible Limit) for upper one-sided intervals or `"LCL"` (i.e. Lower Credible Limit) for lower one-sided intervals.
method_CI	String that indicates if the limits should be computed through the logSMNG quantile function [qlSMNG](option `"exact"`, default), or by randomly generating a sample (`"simulation"`) using the function [rlSMNG].
rel_tol_CI	Level of relative tolerance required for the `integrate` procedure or for the infinite sum. Default set to `1e-5`.
nrep_CI	Number of simulations in case of `method="simulation"`.

Details

The function allows to carry out Bayesian inference for the unconditional quantiles of a sample that is assumed log-normally distributed.

A generalized inverse Gaussian prior is assumed for the variance in the log scale σ^2, whereas a flat improper prior is assumed for the mean in the log scale ξ.

Two alternative hyperparamters setting are implemented (choice controlled by the argument `method`): a weakly informative proposal and an optimal one.

Value

The function returns the prior parameters and their posterior values, summary statistics of the log-scale parameters and the estimate of the specified quantile: the posterior mean and variance are provided by default. Moreover, the user can control the computation of posterior intervals.

Source

Gardini, A., C. Trivisano, and E. Fabrizi. *Bayesian inference for quantiles of the log-normal distribution.* Biometrical Journal (2020).

Examples

```
library(BayesLN)
data("EPA09")
LN_Quant(x = EPA09, x_transf = FALSE, quant = 0.95, method = "optimal", CI = FALSE)
LN_Quant(x = EPA09, x_transf = FALSE, quant = 0.95, method = "weak_inf",
         alpha_CI = 0.05, type_CI = "UCL", nrep_CI = 1e3) # increase nrep_CI
```

LN_QuantReg   *Bayesian estimate of the log-normal conditioned quantiles*

Description

This function produces a point estimate for the log-normal distribution quantile of fixed level quant.

Usage

```
LN_QuantReg(
  y,
  X,
  Xtilde,
  quant,
  method = "weak_inf",
  guess_s2 = NULL,
  y_transf = TRUE,
  CI = TRUE,
  method_CI = "exact",
  alpha_CI = 0.05,
  type_CI = "two-sided",
  rel_tol_CI = 1e-05,
  nrep_CI = 1e+05
)
```

Arguments

y	Vector of observations of the response variable.
X	Design matrix.
Xtilde	Covariate patterns of the units to estimate.
quant	Number between 0 and 1 that indicates the quantile of interest.
method	String that indicates the prior setting to adopt. Choosing "weak_inf" a weakly informative prior setting is adopted, whereas selecting "optimal" the hyperparameters are fixed trough a numerical optimization algorithm aimed at minimizing the frequentist MSE.
guess_s2	Specification of a guess for the variance if available. If not, the sample estimate is used.
y_transf	Logical. If TRUE, the y vector is assumed already log-transformed.
CI	Logical. With the default choice TRUE, the posterior credibility interval is computed.
method_CI	String that indicates if the limits should be computed through the logSMNG quantile function qlSMNG (option "exact", default), or by randomly generating ("simulation") using the function rlSMNG.
alpha_CI	Level of credibility of the posterior interval.
type_CI	String that indicates the type of interval to compute: "two-sided" (default), "UCL" (i.e. Upper Credible Limit) for upper one-sided intervals or "LCL" (i.e. Lower Credible Limit) for lower one-sided intervals.
rel_tol_CI	Level of relative tolerance required for the integrate procedure or for the infinite sum. Default set to 1e-5.
nrep_CI	Number of simulations for the C.I. in case of method="simulation" and for the posterior of the coefficients vector.

Details

The function allows to carry out Bayesian inference for the conditional quantiles of a sample that is assumed log-normally distributed. The design matrix containing the covariate patterns of the sampled units is X, whereas Xtilde contains the covariate patterns of the unit to predict.

The classical log-normal linear mixed model is assumed and the quantiles are estimated as:

$$\theta_p(x) = exp(x^T\beta + \Phi^{-1}(p))$$

.

A generalized inverse Gaussian prior is assumed for the variance in the log scale σ^2, whereas a flat improper prior is assumed for the vector of coefficients β.

Two alternative hyperparamters setting are implemented (choice controlled by the argument method): a weakly informative proposal and an optimal one.

Value

The function returns the prior parameters and their posterior values, summary statistics of the parameters β and σ^2, and the estimate of the specified quantile: the posterior mean and variance are provided by default. Moreover the user can control the computation of posterior intervals.

#'@source

Gardini, A., C. Trivisano, and E. Fabrizi. *Bayesian inference for quantiles of the log-normal distribution.* Biometrical Journal (2020).

NCBC   *Naval Construction Battalion Center data*

Description

Data frame of 17 observations in 2 variables

Usage

NCBC

Format

Dataframe with 2 variables:

al: aluminium concentration measures.

mn: manganese concentration measures.

Source

Singh, Ashok K., Anita Singh, and Max Engelhardt. *The lognormal distribution in environmental applications.* Technology Support Center Issue Paper. (1997).

ReadingTime   *Reading Times data*

Description

Data frame of 547 observations in 4 variables

Usage

ReadingTime

Format

Dataframe with variables:

subj: label indicating the subject.

item: label indicating the item read.

so: variable assuming value 1 (object relative condition) and -1 (subject relative condition).

log_rt: logarithm of the reading time measured.

Source

E. Gibson and H.-H. I. Wu. *Processing chinese relative clauses in context.* Language and Cognitive Processes, 28(1-2):125-155. (2008).

SMNGdistribution *SMNG and logSMNG Distributions*

Description

Density function, distribution function, quantile function and random generator for the SMNG distribution and the logSMNG. It requires the specification of a five prameters vector: mu, delta, gamma, lambda and beta.

Usage

```
dSMNG(
  x,
  mu = 0,
  delta,
  gamma,
  lambda,
  beta = 0,
  inf_sum = FALSE,
  rel_tol = 1e-05
)

pSMNG(q, mu, delta, gamma, lambda, beta, rel_tol = 1e-05)

qSMNG(p, mu, delta, gamma, lambda, beta, rel_tol = 1e-05)

rSMNG(n, mu, delta, gamma, lambda, beta)

dlSMNG(x, mu = 0, delta, gamma, lambda, beta, inf_sum = FALSE, rel_tol = 1e-05)

plSMNG(q, mu, delta, gamma, lambda, beta, rel_tol = 1e-05)

qlSMNG(p, mu, delta, gamma, lambda, beta, rel_tol = 1e-05)

rlSMNG(n, mu, delta, gamma, lambda, beta)
```

Arguments

`x, q`	Vector of quantiles.
`mu`	Location parameter, default set to 0.
`delta`	Concentration parameter, must be positive.
`gamma`	Tail parameter, must be positive.
`lambda`	Shape parameter.
`beta`	Skewness parameter, default set to 0 (symmetric case).
`inf_sum`	Logical: if FALSE (default) the integral representation of the SMNG density is used, otherwise the infinite sum is employed.
`rel_tol`	Level of relative tolerance required for the `integrate` procedure or for the infinite sum convergence check. Default set to `1e-5`.
`p`	Vector of probabilities.
`n`	Sample size.

Details

The SMNG distribution is a normal scale-mean mixture distribution with a GIG as mixing distribution. The density can be expressed as an infinite sum of Bessel K functions and it is characterized by 5 parameters.

Moreover, if X is SMNG distributed, then $Z = exp(X)$ is distributed as a log-SMNG distribution.

Value

`dSMNG` and `dlSMNG` provide the values of the density function at a quantile x for, respectively a SMNG distribution and a log-SMNG.

`pSMNG` and `plSMNG` provide the cumulative distribution function at a quantile q.

`qSMNG` and `qlSMNG` provide the quantile corresponding to a probability level p.

`rSMNG` and `rlSMNG` generate n independent samples from the desired distribution.

Examples

```
### Plots of density and cumulative functions of the SMNG distribution
x<-seq(-10,10,length.out = 500)

plot(x,dSMNG(x = x,mu = 0,delta = 1,gamma = 1,lambda = 1,beta= 2),
    type="l",ylab="f(x)")
lines(x,dSMNG(x = x,mu = 0,delta = 1,gamma = 1,lambda = 1,beta= -2),col=2)
title("SMNG density function")

plot(x,pSMNG(q = x,mu = 0,delta = 1,gamma = 1,lambda = 1,beta= 2),
    type="l",ylab="F(x)")
lines(x,pSMNG(q = x,mu = 0,delta = 1,gamma = 1,lambda = 1,beta= -2),col=2)
title("SMNG cumulative function")
```

```
### Plots of density and cumulative functions of the logSMNG distribution
x<-seq(0,20,length.out = 500)

plot(x,dlSMNG(x = x,mu = 0,delta = 1,gamma = 1,lambda = 2,beta = 1),
     type="l",ylab="f(x)",ylim = c(0,1.5))
lines(x,dlSMNG(x = x,mu = 0,delta = 1,gamma = 1,lambda = 2,beta = -1),col=2)
title("logSMNG density function")

plot(x,plSMNG(q = x,mu = 0,delta = 1,gamma = 1,lambda = 2,beta = 1),
     type="l",ylab="F(x)",ylim = c(0,1))
lines(x,plSMNG(q = x,mu = 0,delta = 1,gamma = 1,lambda = 2,beta = -1),col=2)
title("logSMNG cumulative function")
```

SMNGmoments   *SMNG Moments and Moment Generating Function*

Description

Functions that implement the mean, the generic moments (both raw and centered) and the moment generating function of the SMNG distribution.

Usage

```
SMNG_MGF(
  r,
  mu = 0,
  delta,
  gamma,
  lambda,
  beta = 0,
  inf_sum = FALSE,
  rel_tol = 1e-05
)

meanSMNG(mu, delta, gamma, lambda, beta)

SMNGmoment(j, mu, delta, gamma, lambda, beta, type = "central")
```

Arguments

r	Coefficient of the MGF. Can be viewed also as the order of the logSMNG moments.
mu	Location parameter, default set to 0.
delta	Concentration parameter, must be positive.
gamma	Tail parameter, must be positive.

lambda	Shape parameter.
beta	Skewness parameter, default set to 0 (symmetric case).
inf_sum	Logical: if FALSE (default), the integral representation of the SMNG density is used, otherwise the infinite sum is employed.
rel_tol	Level of relative tolerance required for the `integrate` procedure or for the infinite sum. Default set to `1e-5`.
j	Order of the moment.
type	String that indicate the kind of moment to comupute. Could be `"central"` (default) or `"raw"`.

Details

If the mean (i.e. the first order raw moment) of the SMNG distribution is required, then the function meanSMNG could be use.

On the other hand, to obtain the generic *j*-th moment both `"raw"` or `"centered"` around the mean, the function momentSMNG could be used.

Finally, to evaluate the Moment Generating Function (MGF) of the SMNG distribution in the point r, the function SMNG_MGF is provided. It is defined only for points that are lower then the parameter gamma, and for integer values of r it could also considered as the *r*-th raw moment of the logSMNG distribution. The last function is implemented both in the integral form, which uses the routine integrate, or in the infinite sum structure.

Examples

```
### Comparisons sample quantities vs true values
sample <- rSMNG(n=1000000,mu = 0,delta = 2,gamma = 2,lambda = 1,beta = 2)
mean(sample)
meanSMNG(mu = 0,delta = 2,gamma = 2,lambda = 1,beta = 2)

var(sample)
SMNGmoment(j = 2,mu = 0,delta = 2,gamma = 2,lambda = 1,beta = 2,type = "central")
SMNGmoment(j = 2,mu = 0,delta = 2,gamma = 2,lambda = 1,beta = 2,type = "raw")-
                        meanSMNG(mu = 0,delta = 2,gamma = 2,lambda = 1,beta = 2)^2

mean(exp(sample))
SMNG_MGF(r = 1,mu = 0,delta = 2,gamma = 2,lambda = 1,beta = 2)
```

Index