Package 'degross'

October 13, 2022

Title Density Estimation from GROuped Summary Statistics

Version 0.9.0

Description Estimation of a density from grouped (tabulated) summary statistics evaluated in each of the big bins (or classes) partitioning the support of the variable. These statistics include class frequencies and central moments of order one up to four. The log-density is modelled using a linear combination of penalised B-splines. The multinomial log-likelihood involving the frequencies adds up to a roughness penalty based on the differences in the coefficients of neighbouring B-splines and the log of a root-n approximation of the sampling density of the observed vector of central moments in each class. The so-obtained penalized log-likelihood is maximized using the EM algorithm to get an estimate of the spline parameters and, consequently, of the variable density and related quantities such as quantiles, see Lambert, P. (2021) <arXiv:2107.03883> for details.

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

Imports cubicBsplines, stats

NeedsCompilation no

Author Philippe Lambert [aut, cre] (Université de Liège / Université catholique de Louvain (Belgium))

Maintainer Philippe Lambert <p.lambert@uliege.be>

Repository CRAN

Date/Publication 2021-08-04 14:30:02 UTC

R topics documented:

`ddegross` *Density function based on an object resulting from the estimation procedure in degross.*

Description

Density function based on an object resulting from the estimation procedure in degross.

Usage

```
ddegross(x, degross.fit, phi)
```

Arguments

`x`	Scalar or vector where the fitted density must be evaluated.
`degross.fit`	A degross.object generated using degross and containing the density estimation results.
`phi`	(Optional) vector of spline parameters for the log density (default: `degross.fit$phi` if missing).

Value

A scalar or vector of the same length as `x` containing the value of the fitted density at `x`.

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

`degross.object`, `pdegross`, `qdegross`, `degross`.

Examples

```
## Generate grouped data
sim = simDegrossData(n=1500, plotting=TRUE, choice=2)

## Create a degrossData object
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)

## Estimate the density
obj.fit = degross(obj.data)

## Superpose the fitted density using the <ddegross> function
curve(ddegross(x,obj.fit),add=TRUE,lty="dashed")
legend("topright",lty="dashed",lwd=2,legend="Estimated",box.lty=0, inset=.04)
```

degross *Density estimation from tabulated data with given frequencies and group central moments.*

Description

Estimation of a density from tabulated summary statistics evaluated within each of the big bins (or classes) partitioning the variable support. These statistics include class frequencies and central moments of orders one up to four. The log-density is modelled using a linear combination of penalized B-splines. The multinomial log-likelihood involving the frequencies adds up to a roughness penalty based on differences of neighboring B-spline coefficients and to the log of a root-n approximation of the sampling density of the observed vector of central moments within each class. The so-obtained penalized log-likelihood is maximized using the EM algorithm to get an estimation of the spline parameters and, hence, of the variable density and related quantities such as quantiles, see Lambert (2021) for details.

Usage

```
degross(degross.data,
       phi0 = NULL, tau0 = 1000,
       use.moments = rep(TRUE,4), freq.min = 20, diag.only=FALSE,
       penalize = TRUE,
       aa = 2, bb = 1e-06, pen.order = 3, fixed.tau = FALSE,
       plotting = FALSE, verbose = FALSE, iterlim=20)
```

Arguments

degross.data	A degrossData.object generated by degrossData.
phi0	Starting value for the K-vector ϕ of B-spline parameters specifying the log-density. Default: NULL.
tau0	Starting value for the roughness penalty parameter. Default: 1000.

`use.moments` Vector with 4 logicals indicating which tabulated sample moments to use as soft constraints. Defaults: `rep(TRUE,4)`.

`freq.min` Minimal big bin frequency required to use the corresponding observed moments as soft constraints. Default: `20`.

`diag.only` Logical indicating whether to ignore the off-diagonal elements of the variance-covariance matrix of the sample central moments. Default: FALSE.

`penalize` Logical indicating whether a roughness penalty of order `pen.order` is required (with $\tau \sim G(aa, bb)$). Default: `TRUE`.

`aa` Positive real giving the first parameter in the Gamma prior for `tau`. Default: `2`.

`bb` Positive real giving the second parameter in the Gamma prior for `tau`. Default: `1e-6`.

`pen.order` Integer giving the order of the roughness penalty. Default: `3`.

`fixed.tau` Logical indicating whether the roughness penalty parameter `tau` is fixed. Default: FALSE, implying its estimation.

`plotting` Logical indicating whether an histogram of the data with the estimated density should be plotted. Default: FALSE.

`verbose` Logical indicating whether details on the estimation progress should be displayed. Default: FALSE.

`iterlim` Maximum number of iterations during the M-step. Default: 20.

Value

An object of class `degross` containing several components from the density estimation procedure. Details can be found in `degross.object`. A summary of its content can be printed using `print.degross` or plotted using `plot.degross`.

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

`degross.object`, `ddegross`, `pdegross`, `qdegross`.

Examples

```
## Simulate grouped data
sim = simDegrossData(n=3500, plotting=TRUE,choice=2,J=3)
print(sim$true.density) ## Display density of the data generating mechanism

## Create a degrossData object
obj.data = with(sim, degrossData(Big.bins=Big.bins, freq.j=freq.j, m.j=m.j))
```

```
print(obj.data)

## Estimate the density underlying the grouped data
obj.fit = degross(obj.data)

## Plot the estimated density...
plot(obj.fit)
## ... and compare it with the ('target') density used to simulate the data
curve(sim$true.density(x),add=TRUE,col="red",lwd=2)
legend("topleft",
       legend=c("Observed freq.","Target density","Estimated density"),
       col=c("grey85","red","black"), lwd=c(10,2,2),
       lty=c("solid","solid","dashed"), box.lty=0, inset=.02)
```

degross.object *Object resulting from the estimation of a density from grouped (tabulated) summary statistics*

Description

An object returned by the degross function is a list containing several components resulting from the density estimation procedure.

Value

A degross object is a list containing, after convergence of the EM algorithm :

- lpost & lpost.ni: value of the log-posterior for the complete data based on the expected small bin frequencies n.i at convergence of the EM algorithm.
- lpost.mj : value of the log-posterior for the observed data based on the big bin frequencies freq.j.
- llik.ni: log-likelihood for the complete data based on the estimated small bin frequencies n.i.
- llik.mj : log-likelihood for the observed data based on the big bin frequencies freq.j.
- moments.penalty : log of the joint (asymptotic) density for the observed sample moments.
- penalty : $\log p(\phi|\tau) + \log p(\tau)$.
- Score & Score.mj: score (w.r.t. ϕ) of the log of the observed joint posterior function.
- Score.ni: score (w.r.t. ϕ) of the log-posterior for the complete data based on the expected small bin frequencies n.i at convergence of the EM algorithm.
- Fisher & Fisher.ni: information matrix (w.r.t. ϕ) based on the log-posterior for the complete data based on the expected small bin frequencies n.i at convergence of the EM algorithm.
- Fisher.mj : information matrix (w.r.t. ϕ) based on the log of the observed joint posterior function.

- `M.j` : theoretical moments of the fitted density within a big bin.
- `pi.i` : small bin probabilities (at convergence).
- `ui` : small bin midpoints.
- `delta` : width of the small bins.
- `gamma.j` : big bin probabilities (at convergence).
- `tau` : value of the roughness penalty parameter τ (`tau0` if `fixed.tau`=TRUE, estimated otherwise).
- `phi` : vector with the spline parameters (at convergence).
- `n.i` : small bin frequencies under the estimated density (at convergence).
- `edf` : the effective degrees of freedom (or effective number of spline parameters) (at convergence).
- `aic` : -2*(`llik.mj` + `moments.penalty`) + 2edf. `bic` : -2(`llik.mj` + `moments.penalty`) + $\log(n)$*`edf`.
- `log.evidence` : approximation to the log of $p(\hat{\phi}_\tau, \hat{\tau}|D)\ |\Sigma_\phi|^{(1/2)}$.
- `degross.data` : the degrossData object from which density estimation proceeded.
- `use.moments` : vector of 4 logicals indicating which tabulated sample moments were used as soft constraints during estimation.
- `diag.only` : logical indicating whether the off-diagonal elements of the variance-covariance matrix of the sample central moments were ignored. Default: FALSE.
- `logNormCst` : log of the normalizing constant when evaluating the density.

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

`degross`, `print.degross`, `plot.degross`.

`degrossData` *Creates a degrossData.object from the observed tabulated frequencies and central moments.*

Description

Creates a degrossData.object from the observed tabulated frequencies and central moments.

Usage

```
degrossData(Big.bins, freq.j, m.j, I=300, K=25)
```

Arguments

`Big.bins`	Vector of length `J+1` with the limits of the `J` big bins containing the data used to produce the tabulated statistics.
`freq.j`	The number of data observed within each big bin.
`m.j`	A matrix of dim `J` by 4 giving the first 4 sample central moments within each of the `J` big bins.
`I`	The number of small bins used for quadrature during the normalization of the density during its estimation. Default: `300`.
`K`	The desired number of B-splines in the basis used for density estimation. Default= `25`.

Value

A degrossData.object, i.e. a list containing:

- `small.bins` : a vector of length `I+1` with the small bin limits.
- `ui` : the `I` midpoints of the small bins.
- `delta` : width of the small bins.
- `I` : the number of small bins.
- `B.i` : a matrix of dim `I` by `K` with the B-spline basis evaluated at the small bin midpoints.
- `K` : number of B-splines in the basis.
- `knots` : equidistant knots supporting the B-splines basis.
- `Big.bins` : vector of length `J+1` with the limits of the `J` big bins containing the data used to produce the tabulated statistics.
- `freq.j` : the number of data observed within each big bin.
- `m.j` : a matrix of dim `J` by 4 giving the first 4 sample central moments within each big bin.
- `J` : the number of big bins.
- `small.to.big` : a vector of length `I` indicating to what big bin each element of `ui` belongs.

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

`print.degrossData`, `degross`.

Examples

```
sim = simDegrossData(n=3500, plotting=TRUE)
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)
```

---

degrossData.object *Object generated from grouped summary statistics, including tabulated frequencies and central moments of order 1 up to 4, to estimate the underlying density using* degross.

Description

An object returned by the degrossData function from tabulated frequencies and central moments of order 1 up to 4. It is used in a second step by degross to estimate the underlying density.

Value

A list containing :

- small.bins : a vector of length I+1 with the small bin limits.
- ui : the I midpoints of the small bins.
- delta : width of the small bins.
- I : the number of small bins.
- B.i : a matrix of dim I by K with the B-spline basis evaluated at the small bin midpoints.
- K : number of B-splines in the basis.
- knots : equidistant knots supporting the B-splines basis.
- Big.bins : vector of length J+1 with the limits of the J big bins containing the data used to produce the tabulated statistics.
- freq.j : the number of data observed within each big bin.
- m.j : a matrix of dim J by 4 giving the first 4 sample central moments within each big bin.
- J : the number of big bins.
- small.to.big : a vector of length I indicating to what big bin each element of ui belongs.

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degrossData, print.degrossData

degross_lpost *Log-posterior (with gradient and Fisher information) for given spline parameters, small bin frequencies, tabulated sample moments and roughness penalty parameter. This function is maximized during the M-step of the EM algorithm to estimate the B-spline parameters entering the density specification.*

Description

Log-posterior (with gradient and Fisher information) for given spline parameters, small bin frequencies, tabulated sample moments and roughness penalty parameter. This function is maximized during the M-step of the EM algorithm to estimate the B-spline parameters entering the density specification.

Usage

```
degross_lpost(phi, tau, n.i, degross.data,
                         use.moments = rep(TRUE,4), freq.min = 20, diag.only=FALSE,
                         penalize = TRUE, aa = 2, bb = 1e-6, pen.order = 3)
```

Arguments

phi	Vector of K B-spline parameters ϕ to specify the log-density.
tau	Roughness penalty parameter.
n.i	Small bin frequencies.
degross.data	A degrossData.object created using the degrossData function.
use.moments	Vector with 4 logicals indicating which tabulated sample moments to use as soft constraints. Defaults: `rep(TRUE,4)`.
freq.min	Minimal big bin frequency required to use the corresponding observed moments as soft constraints. Default: `20`.
diag.only	Logical indicating whether to ignore the off-diagonal elements of the variance-covariance matrix of the sample central moments. Default: FALSE.
penalize	Logical indicating whether a roughness penalty of order `pen.order` is required (with $\tau \sim G(aa, bb)$). Default: `TRUE`.
aa	Positive real giving the first parameter in the Gamma prior for `tau`. Default: `2`.
bb	Positive real giving the second parameter in the Gamma prior for `tau`. Default: `1e-6`.
pen.order	Integer giving the order of the roughness penalty. Default: `3`.

Value

A list containing :

- `lpost`, `lpost.ni` : value of the log-posterior based on the given small bin frequencies `n.i` and the tabulated sample moments.

- lpost.mj : value of the log-posterior based on the big bin frequencies degross.data$freq.j and the tabulated sample moments.
- llik.ni : multinomial log-likelihood based on the given small bin frequencies n.i.
- llik.mj : multinomial log-likelihood based on the big bin frequencies degross.data$freq.j.
- moments.penalty : log of the joint (asymptotic) density for the observed sample moments.
- penalty : $\log p(\phi|\tau) + \log p(\tau)$.
- Score, Score.ni : score (w.r.t. ϕ) of lpost.ni.
- Score.mj : score (w.r.t. ϕ) of lpost.mj.
- Fisher & Fisher.ni: information matrix (w.r.t. ϕ) of lpost.ni.
- Fisher.mj : information matrix (w.r.t. ϕ) of lpost.mj.
- M.j : theoretical moments of the density (resulting from ϕ) within a big bin.
- pi.i : small bin probabilities.
- ui : small bin midpoints.
- delta : width of the small bins.
- gamma.j : Big bin probabilities.
- tau : reminder of the value of the roughness penalty parameter τ.
- phi : reminder of the vector of spline parameters (defining the density).
- n.i : reminder of the small bin frequencies given as input.

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degross_lpostBasic, degross, degross.object.

Examples

```
sim = simDegrossData(n=3500, plotting=TRUE,choice=2) ## Generate grouped data
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)
obj.fit = degross(obj.data) ## Estimate the underlying density
## Evaluate the log-posterior at convergence
res = with(obj.fit, degross_lpost(phi, tau, n.i, obj.data, diag.only=diag.only))
print(res$Score) ## Score of the log posterior at convergence
```

degross_lpostBasic

Log-posterior for given spline parameters, big bin (and optional: small bin) frequencies, tabulated sample moments and roughness penalty parameter. Compared to degross_lpost, no Fisher information matrix is computed and the gradient evaluation is optional, with a resulting computational gain.

Description

Log-posterior for given spline parameters, big bin (and optional: small bin) frequencies, tabulated sample moments and roughness penalty parameter. Compared to degross_lpost, no Fisher information matrix is computed and the gradient evaluation is optional, with a resulting computational gain.

Usage

```
degross_lpostBasic(phi, tau, n.i, degross.data,
                         use.moments = rep(TRUE,4), freq.min = 20, diag.only=FALSE,
                             gradient=FALSE,
                             penalize = TRUE, aa = 2, bb = 1e-6, pen.order = 3)
```

Arguments

phi	Vector of K B-spline parameters ϕ to specify the log-density.
tau	Roughness penalty parameter.
n.i	Small bin frequencies.
degross.data	A degrossData.object created using the degrossData function.
use.moments	Vector with 4 logicals indicating which tabulated sample moments to use as soft constraints. Defaults: rep(TRUE,4).
freq.min	Minimal big bin frequency required to use the corresponding observed moments as soft constraints. Default: 20.
diag.only	Logical indicating whether to ignore the off-diagonal elements of the variance-covariance matrix of the sample central moments. Default: FALSE.
gradient	Logical indicating if the gradient (Score) of the $\log p(\phi\|\tau, data)$ should be computed (default: FALSE).
penalize	Logical indicating whether a roughness penalty of order pen.order is required (with $tau \sim G(aa, bb)$). Default: TRUE.
aa	Real giving the first parameter in the Gamma prior for tau. Default: 2.
bb	Real giving the second parameter in the Gamma prior for tau. Default: 1e-6.
pen.order	Integer giving the order of the roughness penalty. Default: 3.

Value

A list containing :

- `lpost.ni` : value of the log-posterior based on the given small bin frequencies `n.i` and the tabulated sample moments.
- `lpost.mj` : value of the log-posterior based on the big bin frequencies `degross.data$freq.j` and the tabulated sample moments.
- `llik.ni` : multinomial log-likelihood based on the given small bin frequencies `n.i`.
- `llik.mj` : multinomial log-likelihood based on the big bin frequencies `degross.data$freq.j` resulting from `n.i`.
- `moments.penalty` : log of the joint (asymptotic) density for the observed sample moments.
- `penalty` : $\log p(\phi|\tau) + \log p(\tau)$.
- `M.j` : theoretical moments of the density (resulting from ϕ) within a big bin.
- `pi.i` : small bin probabilities.
- `ui` : small bin midpoints.
- `delta` : width of the small bins.
- `gamma.j` : big bin probabilities.
- `tau` : reminder of the value of the roughness penalty parameter τ.
- `phi` : reminder of the vector of spline parameters (defining the density).
- `n.i` : reminder of the small bin frequencies given as input.
- `freq.j` : reminder of the big bin frequencies in `degross.data$freq.j`.

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

`degross_lpost`, `degross`, `degross.object`.

Examples

```
sim = simDegrossData(n=3500, plotting=TRUE,choice=2) ## Generate grouped data
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)
obj.fit = degross(obj.data) ## Estimate the underlying density
phi.hat = obj.fit$phi ; tau.hat = obj.fit$tau
## Evaluate the log-posterior at convergence
res = degross_lpostBasic(phi=phi.hat, tau=tau.hat, degross.data=obj.data,
                        gradient=TRUE)
print(res)
```

pdegross *Cumulative distribution function (cdf) based on an object resulting from the estimation procedure in degross.*

Description

Cumulative distribution function (cdf) based on an object resulting from the estimation procedure in degross.

Usage

```
pdegross(x, degross.fit, phi)
```

Arguments

x	Scalar or vector where the fitted cdf must be evaluated.
degross.fit	A degross.object generated using degross and containing the density estimation results.
phi	(Optional) vector of spline parameters for the log density (default: degross.fit$phi if missing).

Value

a scalar or vector of the same length as x containing the value of the fitted cdf at x.

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degross.object, ddegross, qdegross, degross.

Examples

```
## Generate grouped data
sim = simDegrossData(n=3500, plotting=TRUE, choice=2)

## Create a degrossData object
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)

## Estimate the density
```

```
obj.fit = degross(obj.data)

## Superpose the fitted cdf using the <pdegross> function
with(sim, curve(true.cdf(x),min(Big.bins),max(Big.bins),
     col="red",lwd=2, ylab="F(x)"))
curve(pdegross(x,obj.fit),add=TRUE,lty="dashed")
legend("topleft", legend=c("Target cdf","Estimated cdf"), lwd=2,
       lty=c("solid","dashed"), col=c("red","black"), box.lty=0, inset=.04)
```

plot.degross *Plot the density estimate obtained from grouped summary statistics using degross and superpose it to the observed histogram.*

Description

Plot the density estimate corresponding to a degross object and superpose it to the observed histogram.

Usage

```
## S3 method for class 'degross'
plot(x, col="black", lwd=2, lty="dashed", xlab="", ylab="Density", main="",...)
```

Arguments

x	A degross.object generated by degross.
col	Color used for plotting the fitted density.
lwd	Line width for the fitted density curve.
lty	Line type for the the fitted density curve.
xlab	Label on the x-axis.
ylab	Label on the y-axis.
main	Title for the generated graph.
...	Further arguments to be passed to hist.

Value

A histogram based on the observed big bin frequencies with the fitted density superposed.

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degross, degross.object, print.degross.

Examples

```
sim = simDegrossData(n=3500, plotting=TRUE,choice=2) ## Generate grouped data
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)
obj.fit = degross(obj.data) ## Estimate the underlying density
plot(obj.fit) ## Plot the fitted density with the data histogram
```

print.degross *Print a 'degross' object.*

Description

Print a summary of the information contained in a degross.object generated by degross for density estimation from tabulated frequency and central moment data.

Usage

```
## S3 method for class 'degross'
print(x, ...)
```

Arguments

x	A degross.object generated by degross.
...	Possible additional printing options.

Value

Print information on the fitted density corresponding to the degross.object x: the estimated central moments within each class (or big bin) are printed with global fit statistics. A summary of the observed data is also provided: it includes the total sample size, the numbers of small and big bins with their limits in addition to the number of B-splines used for density estimation with degross.

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degross, plot.degross.

Examples

```
sim = simDegrossData(n=3500, plotting=TRUE)
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)

## Estimate the density underlying the grouped data
obj.fit = degross(obj.data)
print(obj.fit)
```

---

print.degrossData *Print a 'degrossData' object.*

Description

Print a summary of the information contained in a degrossData.object used by degross for density estimation from tabulated frequency and moment data.

Usage

```
## S3 method for class 'degrossData'
print(x, ...)
```

Arguments

x	A degrossData.object generated by degrossData.
...	Possible additional printing options for a matrix object.

Value

Print the tabulated summary statistics contained in the degrossData.object x, with additional information on the total sample size, numbers of small and big bins with their limits, the number of B-splines planned for density estimation using degross.

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degross, degrossData

Examples

```
sim = simDegrossData(n=3500, plotting=TRUE)
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)
```

qdegross *Quantile function based on an object resulting from the estimation procedure in degross.*

Description

Quantile function based on an object resulting from the estimation procedure in degross.

Usage

```
qdegross(p, degross.fit, phi, get.se=FALSE, cred.level=.95, eps=1e-4)
```

Arguments

`p`	Scalar or vector of probabilities in (0,1) indicating the requested fitted quantiles Q(p) based on the density estimation results in `degross.fit`.
`degross.fit`	A `degross.object` generated using degross and containing the density estimation results.
`phi`	(Optional) vector of spline parameters for the log density (default: `degross.fit$phi` if missing).
`get.se`	Logical indicating if standard errors for Q(p) are requested (default: FALSE).
`cred.level`	Level of credible intervals for Q(p).
`eps`	Precision with which each quantile should be computed (default: 1e-4).

Value

A scalar or vector `x` of the same length as `p` containing the values Q(p) at which the cdf `pdegross(x,degross.fit)` is equal to `p`. When `get.se` is TRUE, a vector or a matrix containing the quantile estimate(s), standard errors and credible interval limits for Q(p) is provided.

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

degross.object, ddegross, pdegross, degross.

Examples

```
## Generate grouped data
sim = simDegrossData(n=3500, plotting=TRUE, choice=2)

## Create a degrossData object
obj.data = degrossData(Big.bins=sim$Big.bins, freq.j=sim$freq.j, m.j=sim$m.j)
print(obj.data)

## Estimate the density
obj.fit = degross(obj.data)

## Corresponding fitted quantiles
p = c(.01,.05,seq(.1,.9,by=.1),.95,.99) ## Desired probabilities
Q.p = qdegross(p,obj.fit) ## Compute the desired quantiles
print(Q.p) ## Estimated quantiles

## Compute the standard error and a 90% credible interval for the 60% quantile
Q.60 = qdegross(.60,obj.fit,get.se=TRUE,cred.level=.90) ## Compute the desired quantile
print(Q.60) ## Estimated quantile, standard error and credible interval
```

Sigma_fun

Variance-covariance of sample central moments (root-n approximation) given the vector mu with the theoretical moments of order 1 to 8. CAREFUL: the result must be divided by n (= sample size)!

Description

Variance-covariance of sample central moments (root-n approximation) given the vector mu with the theoretical moments of order 1 to 8. CAREFUL: the result must be divided by n (= sample size)!

Usage

```
Sigma_fun(mu)
```

Arguments

`mu` Vector of length 8 with the first 8 theoretical central moments.

Value

Variance-covariance matrix of the first four sample central moments (CAREFUL: a division by the sample size is further required !)

Author(s)

Philippe Lambert <p.lambert@uliege.be>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

Examples

```
mu = numeric(8)
dfun = function(x) dgamma(x,10,5)
mu[1] = integrate(function(x) x*dfun(x),0,Inf)$val
for (j in 2:8) mu[j] = integrate(function(x) (x-mu[1])^j*dfun(x),0,Inf)$val
Sigma_fun(mu)
```

---

simDegrossData *Simulation of grouped data and their sample moments to illustrate the degross density estimation procedure*

Description

Simulation of grouped data and their sample moments to illustrate the degross density estimation procedure

Usage

```
simDegrossData(n, plotting=TRUE, choice=2, J=3)
```

Arguments

n	Desired sample size
plotting	Logical indicating whether the histogram of the simulated data should be plotted. Default: FALSE
choice	Integer in 1:3 indicating from which mixture of distributions to generate the data
J	Number of big bins

Value

A list containing tabulated frequencies and central moments of degrees 1 to 4 for data generated using a mixture density. This list contains :

- `n` : total sample size.
- `J` : number of big bins.
- `Big.bins` : vector of length `J+1` with the big bin limits.

- `freq.j` : vector of length `J` with the observed big bin frequencies.
- `m.j` : `J` by 4 matrix with on each row the observed first four sample central moments within a given big bin.
- `true.density` : density of the raw data generating mechanism (to be estimated from the observed grouped data).
- `true.cdf` : cdf of the raw data generating mechanism (to be estimated from the observed grouped data).

Author(s)

Philippe Lambert <`p.lambert@uliege.be`>

References

Lambert, P. (2021) Moment-based density and risk estimation from grouped summary statistics. arXiv:2107.03883.

See Also

`degrossData`.

Examples

```
## Generate data
sim = simDegrossData(n=3500, plotting=TRUE, choice=2, J=3)
print(sim$true.density) ## Display density of the data generating mechanism

# Create a degrossData object
obj.data = with(sim, degrossData(Big.bins=Big.bins, freq.j=freq.j, m.j=m.j))
print(obj.data)
```

Index