

Package ‘ECG’

July 21, 2025

Type Package

Title Center of Gravity Methods

Version 0.5.2

Date 2024-10-28

Author Hugo Gasca-Aragon [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-5384-2530>>)

Maintainer Hugo Gasca-Aragon <hugo_gasca_aragon@hotmail.com>

Description Implementation of the Centre of Gravity method and the Extrapolated Centre of Gravity method. It supports replicated observations.

Cameron, D.G., et al (1982) <[doi:10.1366/0003702824638610](https://doi.org/10.1366/0003702824638610)>

JCGM (2008) <[doi:10.59161/JCGM100-2008E](https://doi.org/10.59161/JCGM100-2008E)>.

Depends R (>= 4.0), graphics (>= 3.4), stats (>= 3.4), MASS (>= 7.3)

License GPL (>= 3)

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2024-10-29 08:20:05 UTC

Contents

ECG-package	2
assess	3
assess.CGdata	3
assess.CGr	4
assess.ECGdata	5
assess.ECGr	6
CGdata	7
CGr	9
ECGdata	12
ECGr	14
plot.CGdata	17
plot.CGr	18

plot.ECGdata	19
plot.ECGr	20
print.CGdata	21
print.CGr	23
print.ECGdata	24
print.ECGr	25
toString.CGdata	26
toString.CGr	27
toString.ECGdata	28
toString.ECGr	29

Index	31
--------------	-----------

ECG-package	<i>Center of Gravity Methods</i>
-------------	----------------------------------

Description

Estimates the local minimum of a series or readings by the center of gravity and the extrapolated center of gravity.

Details

Package: ECG
Type: Package
Version: 0.5.1
Date: 2024-10-01
License: GPL (>= 3)

Define a series of readings of a series as a data frame object containing: (date of reading, independent variable, replicated observations for control, replicated observations for unknown) Then estimate the minimum by analysing its profile with the CGr method, Optionally estimate the extrapolated minimum by analysing its profile with the ECGr method.

Author(s)

H. Gasca-Aragon
Maintainer: H. Gasca-Aragon <hugo_gasca_aragon@hotmail.com>

References

JCGM 100:2008. *Guide to the expression of uncertainty of measurement*.
Cameron et al. *Precision in Condensed Phase Vibrational Spectroscopy*, Applied Spectroscopy, Vol 36, Number 3, 1982.

assess	<i>Generic method for assessing an estimate against a reference value</i>
--------	---

Description

Generic method for assessing an estimate against a reference value by the En index.

Usage

```
assess(x, y, x.B = list(u=0, dof=Inf), alpha=0.05)
```

Arguments

x	a Estimator object
y	a list with the reference value (x, u, dof)
x.B	a list with the estimate of uncertainty type B for the x object and its degrees of freedom
alpha	numeric, the level of significance used to compute the coverage factor and the expanded uncertainty

Value

the numeric value of the En index evaluation. This is a normalized error value with expanded uncertainty equals 1.0.

Author(s)

H. Gasca-Aragon

assess.CGdata	<i>Evaluates the CGdata object estimation</i>
---------------	---

Description

Evaluates the CGdata object estimation against a reference value by the En index.

Usage

```
## S3 method for class 'CGdata'
assess(x, y, x.B=list(u=0, dof=Inf), alpha=x$input$alpha)
```

Arguments

x	a CGdata object
y	a list with the reference value (x, u)
x.B	a list with the estimate of uncertainty type B for the x object and its degrees of freedom
alpha	a numeric, the level of significance to compute the coverage factor and expanded uncertainty

Value

the numeric value of the En index evaluation. This is a normalized error value with expanded uncertainty equals 1.0.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)

N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

CGres<- CGdata(dat)

X.ref <- list(x=1250, u=0, dof=Inf)

assess(CGres, y=X.ref)
```

assess.CGr

Evaluates the CGr object estimation

Description

Evaluates the CGr object estimation against a reference value by the En index.

Usage

```
## S3 method for class 'CGr'
assess(x, y, x.B = list(u=0, dof=Inf), alpha=x$input$alpha)
```

Arguments

<code>x</code>	a Estimator object
<code>y</code>	a list with the reference value (x, u, dof)
<code>x.B</code>	a list with the estimate of uncertainty type B for the x object and its degrees of freedom
<code>alpha</code>	numeric, the level of significance used to compute the coverage factor and the expanded uncertainty

Value

the numeric value of the En index evaluation.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
X.ref<- list(x=1250, u=0, dof=Inf)
N<- 1000
d0<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)
set.seed(12345)
d1<- d0+rnorm(5/2*N, 0, 0.01)
d2<- d0+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2)))

CGres<- CGr(dat, columns=c(2,3))
assess(CGres, X.ref)
```

assess.ECGdata	<i>Evaluates the ECGdata object estimation</i>
----------------	--

Description

Evaluates the ECGdata object estimation against the reference value by the En index.

Usage

```
## S3 method for class 'ECGdata'
assess(x, y, x.B = list(u=0, dof=Inf), alpha=x$input$alpha)
```

Arguments

x	a Estimator object
y	a list with the reference value (x, u, dof)
x.B	a list with the estimate of uncertainty type B for the x object and its degrees of freedom
alpha	numeric, the level of significance used to compute the coverage factor and the expanded uncertainty

Value

the numeric value of the En index evaluation.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

ECGres<- ECGdata(dat)

X.ref <- list(x=1250, u=0, dof=Inf)
assess(ECGres, y=X.ref)
```

assess.ECGr

Evaluates the ECGr object estimation

Description

Evaluates the ECGr object estimation against a reference value by the En index.

Usage

```
## S3 method for class 'ECGr'
assess(x, y, x.B = list(u=0, dof=Inf), alpha=x$input$alpha)
```

Arguments

x	a Estimator object
y	a list with the reference value (x, u, dof)
x.B	a list with the estimate of uncertainty type B for the x object and its degrees of freedom
alpha	numeric, the level of significance used to compute the coverage factor and the expanded uncertainty

Value

the numeric value of the En index evaluation. This is a normalized error value with expanded uncertainty equals 1.0.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
X.ref<- list(x=pi/2, u=0, dof=Inf)
N<- 1000
d0<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)
set.seed(12345)
d1<- d0+rnorm(5/2*N, 0, 0.01)
d2<- d0+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2)))
ECGres<- ECGr(dat, min(dat$x), max(dat$x), columns=c(2,3),
responseLowerLimit=0, responseUpperLimit=100)
assess(ECGres, X.ref)
```

CGdata

Creates a CGdata object

Description

Builds a CGdata (center of gravity) object for a series of single observations.

Usage

```
CGdata(data, from=min(data$x), to=max(data$x), responseFraction = 0.5,
useConstantDelta = FALSE, fixedResponseFraction = 0.5,
useFixedResponseFraction = FALSE, replaceOutliers = TRUE,
responseLowerLimit = min(data$y), responseUpperLimit = max(data$y),
alpha = 0.05, signifDigits = 2, ...)
```

Arguments

<code>data</code>	a data frame structure containing (x, y) columns.
<code>from</code>	a numeric value with the initial value of x to search for a local minimum.
<code>to</code>	a numeric value with the final value of x to search for a local minimum.
<code>responseFraction</code>	a real value with the fraction (0,1) of the maximum height to be considered in the analysis.
<code>useConstantDelta</code>	a logic value, if true then it assumes the values of x increments at constant rate, otherwise it computes each increment.
<code>fixedResponseFraction</code>	a numeric with the fraction of height to be used as a reference to normalize, default value is 0.5.
<code>useFixedResponseFraction</code>	a logic value, if TRUE then it uses the value of <code>fixedResponseFraction</code> to normalize all the computations, otherwise it uses the value of <code>responseFraction</code> to normalize, default value is TRUE.
<code>replaceOutliers</code>	a logic value, if true then it uses the value of <code>responseLowerLimit</code> and <code>responseUpperLimit</code> to replace outlier values. Default value is TRUE.
<code>responseLowerLimit</code>	a real value to be used as the default to replace outlier values lower than expected.
<code>responseUpperLimit</code>	a real value to be used as the default to replace outlier values larger than expected.
<code>alpha</code>	a real value, it defines the level of error type I used to estimate the coverage factor k_p and the expanded uncertainty, its default value is 0.05.
<code>signifDigits</code>	an integer value, it defines the number of significant digits to be used for displaying the result and its uncertainty, default value is 2.
<code>...</code>	additional parameters.

Value

<code>x</code>	numeric, the estimated value
<code>u</code>	numeric, the estimated uncertainty associated to x
<code>moments</code>	numeric vector, the estimated mean, variance, skewness and kurtosis
<code>input</code>	list, contains the input parameters
<code>frame</code>	list, contains the reference values of the analysis. This information is used to build a verbose version of its plot. The content of the list is: <code>y.x.band.min</code> the local maximum found in the lower region of the analysis region. <code>y.x.band.max</code> the local maximum found in the upper region of the analysis region. <code>x.min.y</code> the value of x where the local minimum y occurs.

`x.max` the value of `x` where the local maximum `y.x.max` occurs.
`x.min` the value of `x` where the local maximum `y.x.min` occurs.
`y.x.max` the maximum height in the upper region of the analysis.
`y.x.min` the maximum height in the lower region of the analysis.
`h` the value of the index of `x` associated with `f.i` fraction of the data in the lower region of analysis.
`k` the value of the index of `x` associated with `f.i` fraction of the data in the upper region of analysis.
`x.h` the value of `x` associated with `f.i` fraction of the data in the lower region of analysis.
`x.k` the value of `x` associated with `f.i` fraction of the data in the upper region of analysis.

`used.data.points` the number of datapoints of `x` used to obtain the estimates, this is equal to `k-h+1`.

Author(s)

H. Gasca-Aragon

Examples

```

require(ECG)

N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

CGres <- CGdata(dat)
CGres

```

CGr

Creates a CGr object

Description

Builds a CGr (center of gravity) object for a series of replicated observations.

Usage

```

CGr(data, from=min(data$x), to=max(data$x), columns, responseFraction = 0.50,
useConstantDelta=FALSE, fixedResponseFraction=0.5, useFixedResponseFraction=FALSE,
replaceOutliers=TRUE, responseLowerLimit=min(data[, columns]),
responseUpperLimit=max(data[, columns]), alpha=0.05,
kp=if(length(columns)<=1) qnorm(1-alpha/2) else
qt(1-alpha/2, length(columns)-1),
signifDigits=2, useRobustStatistics = TRUE, ...)

```

Arguments

<code>data</code>	a data frame structure containing (date, x, y1, ..., yn) columns, it may contain some other columns.
<code>from</code>	a numeric value with the initial value of x to search for a local minimum.
<code>to</code>	a numeric value with the final value of x to search for a local minimum.
<code>columns</code>	a vector of indexes of the columns to be considered in the profile.
<code>responseFraction</code>	numeric, fraction of the maximum height to be considered in the analysis,
<code>useConstantDelta</code>	boolean flag, if true constant increment in the x values is assumed, otherwise the difference is computed for each increment of x.
<code>fixedResponseFraction</code>	a numeric with the fraction of hieght to be used as a reference to normilize.
<code>useFixedResponseFraction</code>	a logical value, if true then it uses the value of f.fixed to normalize all the computations, otherwise it uses the values of extrapolation sequence of fractions to normalize.
<code>replaceOutliers</code>	a logic value, if true then it uses the value of responseLowerLimit and responseUpperLimit to replace outlier values. Default value is TRUE.
<code>responseLowerLimit</code>	a real value to be used as the default to replace outlier values lower than expected, its default value is 0.
<code>responseUpperLimit</code>	a real value to be used as the default to replace outlier values larger than expected, its default value is 1.
<code>alpha</code>	a real value, define the level of significance for building confidence interval.
<code>kp</code>	a real value, it defines the coverage factor to be used to estimate the expanded uncertainty. It is build based on the level of significance alpha and assumes a T distribution of the error terms with the degrees of freedom equals to the number of columns provided minus one, its default value is $qnorm(1-\alpha/2)$ for one column otherwise $qt(1-\alpha/2, \text{length}(\text{columns})-1)$.
<code>signifDigits</code>	number of significant digits used to display the result.
<code>useRobustStatistics</code>	a logical value, if true then median and mad are used to estimate location and dispersion otherwise the mean and standard deviation are used.
<code>...</code>	additional parameters.

Value

<code>x</code>	numeric, the estimated value
<code>u</code>	numeric, the estimated uncertainty associated to x
<code>moments</code>	numeric vector, the estimated mean, variance, skweness and kurtosis

input	list, contains the current input parameters, including the default values additional parameters passed through . . . are not included.
frame	list, contains the reference values of the analysis. This information is used to build a verbose version of its plot. The content of the list is: <code>y.x.band.min</code> the local maximum found in the lower region of the analysis region. <code>y.x.band.max</code> the local maximum found in the upper region of the analysis region. <code>x.min.y</code> the value of <code>x</code> where the local minimum <code>y</code> occurs. <code>x.max</code> the value of <code>x</code> where the local maximum <code>y.x.max</code> occurs. <code>x.min</code> the value of <code>x</code> where the local maximum <code>y.x.min</code> occurs. <code>y.x.max</code> the maximum height in the upper region of the analysis. <code>y.x.min</code> the maximum height in the lower region of the analysis. <code>h</code> the value of the index of <code>x</code> associated with <code>f.i</code> fraction of the data in the lower region of analysis. <code>k</code> the value of the index of <code>x</code> associated with <code>f.i</code> fraction of the data in the upper region of analysis. <code>x.h</code> the value of <code>x</code> associated with <code>f.i</code> fraction of the data in the lower region of analysis. <code>x.k</code> the value of <code>x</code> associated with <code>f.i</code> fraction of the data in the upper region of analysis. <code>used.data.points</code> the number of datapoints of <code>x</code> used to obtain the estimates, this is equal to <code>k-h+1</code> .

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)

N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
d2<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2))
)

CGres<- CGr(dat, columns=c(2,3))
CGres
```

ECGdata	<i>Creates a ECGdata object</i>
---------	---------------------------------

Description

Builds a ECGdata object to estimate an extrapolation of the local minimum in the response for a series of single observations.

Usage

```
ECGdata(data, from=min(data$x), to=max(data$x), useConstantDelta = FALSE,
maxResponseFraction = 0.5, minResponseFraction = 0.05,
byResponseFraction = -0.05, fixedResponseFraction = 0.5,
useFixedResponseFraction = FALSE, replaceOutliers = TRUE,
responseLowerLimit=min(data$y), responseUpperLimit=max(data$y),
alpha = 0.05, kp = qnorm(1-alpha/2), signifDigits = 2,
useRobustStatistics = TRUE, ...)
```

Arguments

<code>data</code>	a data frame structure containing (x, y) columns.
<code>from</code>	a numeric value with the initial value of x to search for a local minimum.
<code>to</code>	a numeric value with the final value of x to search for a local minimum.
<code>useConstantDelta</code>	a logical value, if true then it uses the mean value of the differences in x, otherwise, it uses the differences in x to estimate the expected value. in the analysis.
<code>maxResponseFraction</code>	a real value with the fraction (0,1) of the maximum height to be considered in the analysis.
<code>minResponseFraction</code>	a real value with the fraction (0,1) of the minimum height to be considered in the analysis.
<code>byResponseFraction</code>	a real value with the fraction (0,1) of the decrement of height to be considered in the analysis. The extrapolation analysis uses the sequence: <code>maxResponseFraction</code> , <code>maxResponseFraction+byResponseFraction</code> , ..., <code>minResponseFraction</code>
<code>fixedResponseFraction</code>	a numeric with the fraction of hieght to be used as a reference to normilize.
<code>useFixedResponseFraction</code>	a logical value, if true then it uses the value of <code>f.fixed</code> to normalize all the computations, otherwise it uses the values of extrapolation sequence of fractions to normalize.
<code>replaceOutliers</code>	a logic value, if true then it uses the value of <code>responseLowerLimit</code> and <code>responseUpperLimit</code> to replace outlier values. Default value is TRUE.

<code>responseLowerLimit</code>	a real value to be used as the default to replace outlier values lower than expected, its default value is 0.
<code>responseUpperLimit</code>	a real value to be used as the default to replace outlier values larger than expected, its default value is 1.
<code>alpha</code>	a real value, define the level of significance for building confidence interval.
<code>kp</code>	a real value, it defines the coverage factor to be used to estimate the expanded uncertainty. It is build based on the level of significance <code>alpha</code> and assumes normal distribution of the error terms, its default value is <code>qnorm(1-alpha/2)</code> .
<code>signifDigits</code>	number of significant digits used to display the result.
<code>useRobustStatistics</code>	a logical value, if true then median and mad are used to estimate location and dispersion otherwise the mean and standard deviation are used.
<code>...</code>	additional parameters.

Details

The data at each step is a subset of the previous step hence the estimates are correlated. However by specifying `useFixedResponseFraction=FALSE` they are normalized against distinct fraction of the height.

Value

<code>x</code>	numeric, the estimated value
<code>u</code>	numeric, the estimated uncertainty associated to <code>x</code>
<code>input</code>	list, contains the input parameters
<code>frame</code>	list, contains the reference values of the analysis. This information is used to build a verbose version of its plot. The content of the list is: <code>kp</code> the updated coverage factor considering the reduced degrees of freedom from using the model used. <code>y.x.band.min</code> the local maximum found in the lower region of the analysis region. <code>x.max</code> the value of <code>x</code> where the local maximum <code>y.x.max</code> occurs. <code>x.min</code> the value of <code>x</code> where the local maximum <code>y.x.min</code> occurs. <code>solutions</code> a matrix with the solutions found for each analyzed fraction of the data. The contained columns are: the location estimate, the standard uncertainty, the response fraction used, the value of the minimum response in the considered band, the maximum response in the considered band, and the number of data points used. <code>type</code> an integer representing the degree of the polynomial model (0=constant model, 1=polynomial model of first degree, 2=polynomial model of second degree). <code>model</code> a linear model summary object.

Author(s)

H. Gasca-Aragon

See AlsoSee Also as [CGdata](#), [print.ECGrdata](#), [plot.ECGrdata](#)**Examples**

```
require(ECGr)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

ECGrdata(dat)
ECGrdata
```

ECGr

*Creates an ECGr object***Description**

Builds an ECGr object to estimate an extrapolation of the local minimum in the response for a series of replicated observations.

Usage

```
ECGr(data, from=min(data$x), to=max(data$x), columns, useConstantDelta=FALSE,
maxResponseFraction=0.5, minResponseFraction=0.05,
byResponseFraction=-0.05, fixedResponseFraction=0.5,
useFixedResponseFraction = FALSE, replaceOutliers = TRUE,
responseLowerLimit = min(data[, columns]),
responseUpperLimit = max(data[, columns]),
alpha=0.05, kp=if(length(columns)<=1) qnorm(1-alpha/2) else
qt(1-alpha/2, length(columns)-1),
signifDigits = 2, useRobustStatistics=TRUE, ...)
```

Arguments

data	a data frame structure containing (date, x, y1, ..., yn) columns, it may contain some other columns.
from	a numeric value with the initial value of x to search for a local minimum.
to	a numeric value with the final value of x to search for a local minimum.
columns	a vector of indexes of the columns to be considered in the profile.

<code>useConstantDelta</code>	a logical value, if true then it uses the mean value of the differences in x, otherwise, it uses the differences in x to estimate the expected value. in the analysis.
<code>maxResponseFraction</code>	a real value with the fraction (0,1) of the maximum height to be considered in the analysis.
<code>minResponseFraction</code>	a real value with the fraction (0,1) of the minimum height to be considered in the analysis.
<code>byResponseFraction</code>	a real value with the fraction (0,1) of the decrement of height to be considered in the analysis. The extrapolation analysis uses the sequence: <code>maxResponseFraction</code> , <code>maxResponseFraction+byResponseFraction</code> , ..., <code>minResponseFraction</code>
<code>fixedResponseFraction</code>	a numeric with the fraction of height to be used as a reference to normalize.
<code>useFixedResponseFraction</code>	a logical value, if true then it uses the value of <code>f.fixed</code> to normalize all the computations, otherwise it uses the values of extrapolation sequence of fractions to normalize.
<code>replaceOutliers</code>	a logic value, if true then it uses the value of <code>responseLowerLimit</code> and <code>responseUpperLimit</code> to replace outlier values. Default value is TRUE.
<code>responseLowerLimit</code>	a real value to be used as the default to replace outlier values lower than expected, its default value is 0.
<code>responseUpperLimit</code>	a real value to be used as the default to replace outlier values larger than expected, its default value is 1.
<code>alpha</code>	a real value, define the level of significance for building confidence interval.
<code>kp</code>	a real value, it defines the coverage factor to be used to estimate the expanded uncertainty. It is build based on the level of significance <code>alpha</code> and assumes normal distribution of the error terms, its default value is <code>qnorm(1-alpha/2)</code> .
<code>signifDigits</code>	number of significant digits used to display the result.
<code>useRobustStatistics</code>	a logical value, if true then median and mad are used to estimate location and dispersion otherwise the mean and standard deviation are used.
<code>...</code>	additional parameters.

Value

<code>x</code>	numeric, the estimated value
<code>u</code>	numeric, the estimated uncertainty associated to x
<code>input</code>	list, contains the input parameters

`frame` list, contains the reference values of the analysis. This information is used to build a verbose version of its plot. The content of the list is:

- `y` average of the response series. Depending on the `useRobustStatistics` value the average can be the mean or the median of the series indicated in the `columns` parameter.
- `u.y` average uncertainty of the response series. Depending on the `useRobustStatistics` value the average can be the standard deviation or the median absolute deviation of the series indicated in the `columns` parameter.
- `kp` the updated coverage factor considering the reduced degrees of freedom from using the model used.
- `x.summary` the estimated location from the average of the series.
- `u.x.summary` the estimated uncertainty associated to the estimated location from the average of the series.

`details` the matrix containing the results for considered fractions in the analysis. The columns are:

- the estimated location, the estimated uncertainty, the minimum response value found,
- the minimum value of the location estimates, the maximum value of the location estimates,
- the estimated coverage factor.
- `used.data.points` the number of data points used in the estimations.

Author(s)

H. Gasca-Aragon

See Also

See Also as [ECGdata](#), [print.ECGr](#), [plot.ECGr](#)

Examples

```
require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
d2<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2))
)
ECGres<- ECGr(dat, columns=c(2,3))
ECGres
```

plot.CGdata	<i>Plots a CGdata object</i>
-------------	------------------------------

Description

Creates a plot of a series of single observations.

Usage

```
## S3 method for class 'CGdata'
plot(x, from=x$input$from, to=x$input$to, xlab = expression(nu(cm^-1)),
     ylab = "Transmittance", add = FALSE, verbose = FALSE, ...)
```

Arguments

x	a CGdata object
from	lower limit point on the predictor value to be plotted
to	upper limit point on the predictor value to be plotted
xlab	a string or expression with the label for the x axis
ylab	a string or expression with the label for the y axis
add	a logical value, if true then the plot is added to an existing one, otherwise it creates a new plot
verbose	a logical value, if true then the plot displays intermediate steps to estimate the extreme point located between the from and to values
...	additional parameters

Value

No return value, called for graphical display side effects

Author(s)

H. Gasca-Aragon

See Also

See Also as [CGdata](#), [print.CGdata](#)

Examples

```
require(ECG)

N<- 1000

set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
```

```

dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

CGres<- CGdata(dat)

plot(CGres, min(dat$x), max(dat$x), verbose = TRUE)

```

plot.CGr

Plots a CGr object

Description

Creates a plot of a series of replicated data. It shows the summary of the series. Optionally it shows the limits of the range of analysis and the estimated local minimum computed as the center of gravity.

Usage

```

## S3 method for class 'CGr'
plot(x, ...)

```

Arguments

x	a CGr object
...	additional parameters

Value

No return value, called for graphical display.

Author(s)

H. Gasca-Aragon

See Also

See Also as [CGr](#), [print.CGr](#)

Examples

```

require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
d2<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),

```

```

y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2))
)
CGres<- CGr(dat, columns=c(2,3))
plot(CGres)

```

plot.ECGdata

*Plots an ECGdata object***Description**

Creates a plot for a series of single observations with the extrapolated center of gravity method.

Usage

```

## S3 method for class 'ECGdata'
plot(x, xlim=range(x$frame$solution[,1]), ylim=c(0, max(x$frame$solution[,5])),
     xlab=expression(nu[i]), ylab=expression(f[i]), add=TRUE, ...)

```

Arguments

x	an ECGdata object
xlim	a range with the x limits for plotting purposes
ylim	a range with the y limits for plotting purposes
xlab	a string or expression with the label for the x axis
ylab	a string or expression with the label for the y axis
add	a logical value, if true then the plot is added to an existing one, otherwise it creates a new plot.
...	additional parameters

Details

It plots the sequence of approximations for each fraction of the dataset, and it shows the estimated value while extrapolated at zero fraction. Use it in collaboration with CGdata::plot to show the data, the search frame, and the sequence of approximations.

Value

No return value.

Author(s)

H. Gasca-Aragon

See Also

See Also as [ECGdata](#), [print.ECGdata](#), [plot.ECGdata](#)

Examples

```

require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

CGres <- CGdata(dat)
ECGres<- ECGdata(dat)

# use it alone, plot the sequence of approximations
plot(ECGres, add = FALSE)

# use it in collaboration with CGdata
# display the Center of Gravity approach for single observation
plot(CGres, min(dat$x), max(dat$x))
# then add the extrapolation sequence
plot(ECGres)

```

plot.ECGr

Plots an ECGr object

Description

Creates a plot for a series of replicated observations with the extrapolated center of gravity method.

Usage

```

## S3 method for class 'ECGr'
plot(x, add = TRUE, ...)

```

Arguments

x	an ECGr object.
add	a logical value, if true then the plot is added to an existing one, otherwise it creates a new plot.
...	additional parameters.

Details

It plots the sequence of approximations for each fraction of the dataset, and it shows the estimated value while extrapolated at zero fraction.

Value

No return value. Used for graphical display.

Author(s)

H. Gasca-Aragon

See Also

See Also as [ECGr](#), [print.ECGr](#)

Examples

```
require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
d2<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2))
)

# used alone
ECGres<- ECGr(dat, columns=c(2,3))
plot(ECGres, add = FALSE)

# used in colaboration with CGr
CGres<- CGr(dat, columns=c(2,3))

# display the Center of Gravity approach for replicated observations
plot(CGres, verbose = TRUE)
# then add the extrapolation sequence
plot(ECGres)
```

print.CGdata

Displays the content of a CGdata object

Description

Displays the content of a CGdata object.

Usage

```
## S3 method for class 'CGdata'
print(x, signifDigits=x$input$signifDigits, alpha=x$input$alpha, verbose=FALSE, ...)
```

Arguments

<code>x</code>	a CGdata object.
<code>signifDigits</code>	a numeric value for the number of significant digits.
<code>alpha</code>	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for <code>x\$frame\$used.data.points-1</code> degrees of freedom.
<code>verbose</code>	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.
<code>...</code>	additional parameters

Value

No return value, called for displaying purposes/side effect.

Author(s)

H. Gasca-Aragon

See Also

objects to See Also as [CGdata](#),

Examples

```
require(ECG)

N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N)))/N*pi-pi*3/4+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

CGres<- CGdata(dat)
CGres
print(CGres)
```

print.CGr	<i>Displays the content of a CGr object</i>
-----------	---

Description

Displays the content of a CGr object.

Usage

```
## S3 method for class 'CGr'
print(x, signifDigits=x$input$signifDigits,
      alpha = x$input$alpha, verbose=FALSE, ...)
```

Arguments

x	a CGr object.
signifDigits	a numeric value for the number of significant digits.
alpha	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for <code>x\$frame\$used.data.points-1</code> degrees of freedom.
verbose	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.
...	additional parameters

Value

No return value.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
d2<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
                 y1=100*(d1-min(d1))/(max(d1)-min(d1)),
                 y2=100*(d2-min(d2))/(max(d2)-min(d2)))
```

```
)
CGres<- CGr(dat, columns=c(2,3))
print(CGres)
```

print.ECGdata	<i>Displays the content of a ECGdata object</i>
---------------	---

Description

Displays the content of a ECGdata object.

Usage

```
## S3 method for class 'ECGdata'
print(x, signifDigits=x$input$signifDigits, alpha = x$input$alpha, verbose=FALSE, ...)
```

Arguments

x	an ECGdata object.
signifDigits	a numeric value for the number of significant digits.
alpha	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for <code>x\$frame\$used.data.points-1</code> degrees of freedom.
verbose	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.
...	additional parameters

Value

No return value. Used for textual displaying side effect.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

ECGres<- ECGdata(dat)
ECGres
print(ECGres)
```

print.ECGr

Displays the content of a ECGr object

Description

Displays the content of a ECGr object.

Usage

```
## S3 method for class 'ECGr'
print(x, signifDigits=x$input$signifDigits,
alpha = x$input$alpha, verbose=FALSE, ...)
```

Arguments

x	an ECGr object.
signifDigits	a numeric value for the number of significant digits.
alpha	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for <code>x\$frame\$used.data.points-1</code> degrees of freedom.
verbose	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.
...	additional parameters

Value

No return value. Used for textual display purposes.

Author(s)

H. Gasca-Aragon

Examples

```

require(ECG)
N<- 1000
set.seed(12345)
d1<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
d2<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2))
)

ECGres<- ECGr(dat, columns=c(2,3))
ECGres
print(ECGres)

CGres<- CGr(dat, columns=c(2,3))
CGres
print(CGres)

```

toString.CGdata

*Converts a CGdata object into a string description***Description**

Converts a CGdata object into a string description for displaying purposes.

Usage

```

## S3 method for class 'CGdata'
toString(x, signifDigits=x$input$signifDigits, alpha=x$input$alpha, verbose=FALSE, ...)

```

Arguments

x	a CGdata object
signifDigits	a numeric value for the number of significant digits.
alpha	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for <code>x\$frame\$used.data.points-1</code> degrees of freedom.
verbose	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.

... additional parameters

Value

a string description of the CGdata object.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
N<- 1000
d0<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)
set.seed(12345)
d1<- d0+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

CGres<- CGdata(dat)
str.res <- toString(CGres)
print(nchar(str.res))
print(str.res)
```

toString.CGr

Converts a CGr object into a string description

Description

Converts a CGr object into a string description for displaying purposes.

Usage

```
## S3 method for class 'CGr'
toString(x, signifDigits=x$input$signifDigits, alpha=x$input$alpha, verbose=FALSE, ...)
```

Arguments

x	a CGr object
signifDigits	a numeric value for the number of significant digits.
alpha	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for $x\$frame\$used.data.points-1$ degrees of freedom.

verbose a boolean, if FALSE the description of the form
 (value, expanded uncertainty) is provided. if TRUE the description is comple-
 mented providing:
 the estimated coverage factor,
 the estimated degrees of freedom,
 the estimated coverage probability.

... additional parameters

Value

a string description of the CGr object.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
X.ref<- list(x=pi/2, u=0)
N<- 1000
d0<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)

set.seed(12345)
d1<- d0+rnorm(5/2*N, 0, 0.01)
d2<- d0+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2)))

CGres<- CGr(dat, columns=c(2,3))
str.res <- toString(CGres)
print(nchar(str.res))
```

toString.ECGdata	<i>Converts a ECGdata object into a string description</i>
------------------	--

Description

Converts a ECGdata object into a string description for displaying purposes.

Usage

```
## S3 method for class 'ECGdata'
toString(x, signifDigits=x$input$signifDigits, alpha=x$input$alpha, verbose=FALSE, ...)
```

Arguments

<code>x</code>	an ECGdata object
<code>signifDigits</code>	a numeric value for the number of significant digits.
<code>alpha</code>	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for <code>x\$frame\$used.data.points-1</code> degrees of freedom.
<code>verbose</code>	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.
<code>...</code>	additional parameters

Value

a string description of the ECGdata object.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
X.ref<- list(x=pi/2, u=0)
N<- 1000
d0<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)

set.seed(12345)
d1<- d0+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y=100*(d1-min(d1))/(max(d1)-min(d1)))

ECGres<- ECGdata(dat)
str.res <- toString(ECGres)
print(str.res)
```

toString.ECGr

Converts an ECGr object into a string description

Description

Converts an ECGr object into a string description for displaying purposes.

Usage

```
## S3 method for class 'ECGr'
toString(x, signifDigits=x$input$signifDigits, alpha=x$input$alpha,
verbose=FALSE, ...)
```

Arguments

x	a ECGr object
signifDigits	a numeric value for the number of significant digits.
alpha	a probability value taken as the significance level for building a symmetric confidence interval assuming a t distribution for $x\$frame\$used.data.points-1$ degrees of freedom.
verbose	a boolean, if FALSE the description of the form (value, expanded uncertainty) is provided. if TRUE the description is complemented providing: the estimated coverage factor, the estimated degrees of freedom, the estimated coverage probability.
...	additional parameters

Value

a string description of the ECGr object.

Author(s)

H. Gasca-Aragon

Examples

```
require(ECG)
X.ref<- list(x=pi/2, u=0)
N<- 1000
d0<- 1-sin(seq(1:(5/2*N))/N*pi-pi*3/4)

set.seed(12345)
d1<- d0+rnorm(5/2*N, 0, 0.01)
d2<- d0+rnorm(5/2*N, 0, 0.01)
dat<- data.frame(x=1:length(d1),
y1=100*(d1-min(d1))/(max(d1)-min(d1)),
y2=100*(d2-min(d2))/(max(d2)-min(d2)))

ECGres<- ECGr(dat, columns=c(2,3))
str.res <- toString(ECGres)
print(nchar(str.res))
```

Index

- * **CGdata**
 - assess.CGdata, 3
 - CGdata, 7
 - CGr, 9
 - plot.CGdata, 17
 - print.CGdata, 21
 - toString.CGdata, 26
- * **CGr**
 - assess.CGr, 4
 - CGr, 9
 - plot.CGr, 18
 - print.CGr, 23
 - toString.CGr, 27
- * **ECGdata**
 - assess.ECGdata, 5
 - ECGdata, 12
 - ECGr, 14
 - plot.ECGdata, 19
 - print.ECGdata, 24
 - toString.ECGdata, 28
- * **ECGr**
 - assess.ECGr, 6
 - ECGr, 14
 - plot.ECGr, 20
 - print.ECGr, 25
 - toString.ECGr, 29
- * **En performance index**
 - assess, 3
 - assess.CGdata, 3
 - assess.CGr, 4
 - assess.ECGdata, 5
 - assess.ECGr, 6
- * **asses**
 - assess, 3
- * **plot**
 - plot.CGdata, 17
 - plot.CGr, 18
 - plot.ECGr, 20
- * **print**
 - print.CGdata, 21
 - print.CGr, 23
 - print.ECGdata, 24
 - print.ECGr, 25
- * **toString**
 - toString.CGdata, 26
 - toString.CGr, 27
 - toString.ECGdata, 28
 - toString.ECGr, 29
- assess, 3
- assess.CGdata, 3
- assess.CGr, 4
- assess.ECGdata, 5
- assess.ECGr, 6
- CGdata, 7, 14, 17, 22
- CGr, 9, 18
- ECG (ECG-package), 2
- ECG-package, 2
- ECGdata, 12, 16, 19
- ECGr, 14, 21
- plot.CGdata, 17
- plot.CGr, 18
- plot.ECGdata, 14, 19, 19
- plot.ECGr, 16, 20
- print.CGdata, 17, 21
- print.CGr, 18, 23
- print.ECGdata, 14, 19, 24
- print.ECGr, 16, 21, 25
- toString.CGdata, 26
- toString.CGr, 27
- toString.ECGdata, 28
- toString.ECGr, 29