Package ‘FuzzySTs’

November 23, 2020

Title Fuzzy Statistical Tools

Description The main goal of this package is to present various fuzzy statistical tools. It intends to provide an implementation of the theoretical and empirical approaches presented in the thesis entitled "The signed distance measure in fuzzy statistical analysis. Some theoretical, empirical and programming advances" (Thesis to be published soon. For the theoretical approaches, see Berkachy R. and Donze L. (2019) <doi:10.1007/978-3-030-03368-2_1>. For the empirical approaches, see Berkachy R. and Donze L. (2016) <ISBN: 978-989-758-201-1>). Important (non-exhaustive) implementation highlights of this package are as follows: (1) a numerical procedure to estimate the fuzzy difference and the fuzzy square. (2) two numerical methods of fuzzification. (3) a function performing different possibilities of distances, including the signed distance and the generalized signed distance for instance. (4) numerical estimations of fuzzy statistical measures such as the variance, the moment, etc. (5) two methods of estimation of the bootstrap distribution of the likelihood ratio in the fuzzy context. (6) an estimation of a fuzzy confidence interval by the likelihood ratio method. (7) testing fuzzy hypotheses and/or fuzzy data by fuzzy confidence intervals in the Kwakernaak - Kruse and Meyer sense. (8) a general method to estimate the fuzzy p-value with fuzzy hypotheses and/or fuzzy data. (9) a method of estimation of global and individual evaluations of linguistic questionnaires. (10) numerical estimations of multi-ways analysis of variance models in the fuzzy context. The unbalance in the considered designs are also foreseen.

Version 0.2

Author Redina Berkachy <redina.berkachy@unifr.ch>, Laurent Donze <laurent.donze@unifr.ch>

Maintainer Redina Berkachy <redina.berkachy@unifr.ch>

Depends R (>= 3.0.0), FuzzyNumbers, polynom

License MIT + file LICENSE

LazyData true

RoxygenNote 7.1.1

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2020-11-23 13:50:03 UTC
R topics documented:

adjusted.weight.MI .................................................. 4
adjusted.weight.SI .................................................. 5
Bertoluzza ......................................................... 6
boot.mean.algo1 .................................................... 7
boot.mean.algo2 .................................................... 8
boot.mean.ml ....................................................... 9
cube ............................................................... 10
D2 ................................................................. 11
Defuzz.FANOVA .................................................... 11
Delta.pq ............................................................ 12
Delta_jki ............................................................. 13
distance ............................................................. 14
DSGD ............................................................... 15
DSGD.G ............................................................. 16
FANOVA ............................................................. 16
FANOVA.approximation ........................................... 18
FANOVA.distance .................................................. 19
FANOVA.exact ..................................................... 20
FANOVA.summary .................................................. 21
ci.ml ............................................................... 22
ci.ml.boot .......................................................... 23
FMANOVA .......................................................... 24
FMANOVA.approximation ........................................ 26
FMANOVA.distance ................................................ 27
FMANOVA.exact .................................................... 28
FMANOVA.interaction.summary ................................ 29
FMANOVA.summary ................................................ 30
Ftests ............................................................... 30
FTukeyHSD ........................................................ 31
FUZZ ............................................................... 32
Fuzzy.CI.ML.test ................................................... 33
Fuzzy.CI.test ....................................................... 35
Fuzzy.decisions .................................................... 37
Fuzzy.decisions.ML ................................................ 38
Fuzzy.Difference ................................................... 40
Fuzzy.exact.variance .............................................. 41
Fuzzy.exact.variance.poly.left ............................... 42
Fuzzy.exact.variance.poly.right .............................. 42
Fuzzy.p.value ....................................................... 43
Fuzzy.p.value.mean ............................................... 44
fuzzy.predicted.values .......................................... 46
fuzzy.residuals ..................................................... 47
Fuzzy.sample.mean ............................................... 47
Fuzzy.sample.variance.approximation ....................... 48
Fuzzy.sample.variance.approximation1 ...................... 48
Fuzzy.sample.variance.approximation2 ...................... 49
R topics documented:

- Fuzzy.sample.variance.approximation3 ........................................ 49
- Fuzzy.sample.variance.approximation4 ........................................ 50
- Fuzzy.sample.variance.approximation5 ........................................ 50
- Fuzzy.Square ................................................................. 51
- Fuzzy.Square.poly.left ...................................................... 51
- Fuzzy.Square.poly.right ..................................................... 52
- Fuzzy.variance .............................................................. 53
- GaussianBellFuzzyNumber .................................................... 54
- GaussianFuzzyNumber ........................................................ 55
- GFUZZ ........................................................................... 56
- GLOB.EVAL ....................................................................... 57
- GLOB.EVAL.mean ............................................................... 59
- GSGD ............................................................................. 61
- IND.EVAL ................................................................. 62
- int.0 ................................................................. 64
- int.ct ............................................................... 65
- int.simpson .......................................................... 65
- int.t ............................................................. 66
- integrate.num ......................................................... 66
- is.alphacuts ............................................................... 67
- is.balanced ............................................................... 67
- is.fuzzification ............................................................. 68
- is.trfuzzification ........................................................... 68
- Kurtosis ..................................................................... 69
- Mid.Spr .................................................................... 70
- Moment ..................................................................... 71
- nbreakpoints ............................................................. 72
- n_jk................................................................. 73
- n_jkq ............................................................. 73
- p.value.fisher .............................................................. 74
- p.value.log ............................................................ 76
- p.value.mean.log ........................................................ 77
- p.value.mean.normal ...................................................... 78
- p.value.mean.poisson .................................................... 80
- p.value.mean.Student .................................................... 81
- p.value.normal .......................................................... 83
- p.value.poisson ........................................................... 84
- p.value.Student .......................................................... 86
- R .......................................................................... 87
- Rho1 ..................................................................... 88
- Rho2 ..................................................................... 89
- Rhop ..................................................................... 89
- Ri .......................................................................... 90
- Sample.variance ............................................................ 91
- SEQ.ORDERING .......................................................... 92
- SEQ.ORDERING.APPROXIMATION .............................................. 92
- SEQ.ORDERING.EXACT ..................................................... 93
- SGD ................................................................. 93
adjusted.weight.MI

Calculates the adjusted weight for a given main-item of a linguistic questionnaire

Description

Calculates the adjusted weight for a given main-item of a linguistic questionnaire

Usage

adjusted.weight.MI(x, i, j, b_j, b_jk, SI)

Arguments

- `x`: the data set to evaluate.
- `i`: an observation index.
- `j`: a main-item index.
- `b_j`: an array referring to the initial weights given to each main-item of the considered main-item. This array will be afterwards re-calculated.
- `b_jk`: a matrix of length(b_j) rows and max(SI) columns expressing the initial weights of each sub-item of a given main-item.
- `SI`: an array representing the total numbers of sub-items per main-item.

Value

A numerical value giving the readjusted weight of the main-item j for the observation i.

Examples

data <- as.data.frame(data)
MI <- 2
SI1 <- 2
SI2 <- 2
SI <- c(SI1,SI2)
b_j <- c(1/2,1/2)
b_jk <- matrix(c(0.5,0.5,0.5,0.5),nrow=2)
PA11 <- c(1,2,3,4,5)
PA12 <- c(1,2,3,4,5)
adjusted.weight.SI

Calculates the adjusted weight for a given sub-item of a linguistic questionnaire

Description

Calculates the adjusted weight for a given sub-item of a linguistic questionnaire

Usage

adjusted.weight.SI(x, i, k, b_jk)

Arguments

x
the data set to evaluate.
i
an observation index.
\( k \)           a sub-item index.
\( b_{jk} \)     an array referring to the initial weights given to each sub-item of the considered
main-item. This array will be afterwards re-calculated.

Value

A numerical value giving the readjusted weight of the sub-item \( k \) of the considered
main-item for the observation \( i \).

Examples

```r
data <- matrix(c(3,4,2,3,2,4,3,4,3,4,4,2,5,3,4,4,3,3,4,4,3,3,4,4,3,
                  3,3,4,3,3,3,4,4,3,3,5,3,4,3,3,3,3), ncol = 4)
adjusted.weight.SI(data, 7, 1, c(0.5,0.5))
```

---

**Bertoluzza**

Calculates a distance by the \( d_{Bertoluzza} \) between fuzzy numbers

Description

Calculates a distance by the \( d_{Bertoluzza} \) between fuzzy numbers

Usage

```r
Bertoluzza(X, Y, i = 1, j = 1, theta = 1/3, breakpoints = 100)
```

Arguments

- **X**         a fuzzy number.
- **Y**         a fuzzy number.
- **i**        parameter of the density function of the Beta distribution, fixed by default to \( i = 1 \).
- **j**        parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).
- **theta**    a numerical value between 0 and 1, representing a weighting parameter. By
default, \( \theta \) is fixed to \( 1/3 \) referring to the Lebesgue space. This measure is
used in the calculations of the following distances: \( d_{Bertoluzza}, d_{mid/spr} \)
and \( d_{phi-wabl/ldev/rdev} \).
- **breakpoints** a positive arbitrary integer representing the number of breaks chosen to build
the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.
boot.mean.algo1

Estimates the bootstrap distribution of the likelihood ratio LR by the Algorithm 1 using the mean

Description

Estimates the bootstrap distribution of the likelihood ratio LR by the Algorithm 1 using the mean

Usage

```r
boot.mean.algo1(
  data.fuzzified, 
  distribution, 
  sig, 
  nsim = 100, 
  mu = NA, 
  sigma = NA, 
  step = 0.1, 
  margin = c(5, 5), 
  breakpoints = 100, 
  plot = TRUE
)
```

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
distribution a distribution chosen between "normal", "poisson", "Student" or "Logistic". 
sig a numerical value representing the significance level of the test. 
nsim an integer giving the number of replications needed in the bootstrap procedure. It is set to 100 by default. 
mu if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA. 
sigma if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA. 
step a numerical value fixed to 0.1, defining the step of iterations on the interval [t-5; t+5]. 
margin an optional numerical couple of values fixed to [5; 5], representing the range of calculations around the parameter t. 
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default. 
plot fixed by default to "FALSE". plot="FALSE" if a plot of the fuzzy number is not required.
Value

Returns a vector of decimals representing the bootstrap distribution of LR.

\[\text{boot.mean.algo2} \]

Estimates the bootstrap distribution of the likelihood ratio LR by the Algorithm 2 using the mean

Description

Estimates the bootstrap distribution of the likelihood ratio LR by the Algorithm 2 using the mean

Usage

\[
\text{boot.mean.algo2}( \text{data.fuzzified}, \text{distribution}, \text{sig}, \text{nsim} = 100, \text{mu} = \text{NA}, \text{sigma} = \text{NA}, \text{step} = 0.1, \text{margin} = \text{c}(5, 5), \text{breakpoints} = 100, \text{plot} = \text{TRUE})
\]

Arguments

data.fuzzified: a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
distribution: a distribution chosen between "normal", "poisson", "Student" or "Logistic".
sig: a numerical value representing the significance level of the test.
nsim: an integer giving the number of replications needed in the bootstrap procedure. It is set to 100 by default.
mu: if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.
sigma: if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA.
step: a numerical value fixed to 0.1, defining the step of iterations on the interval \[t-5; t+5\].
margin: an optional numerical couple of values fixed to \[5; 5\], representing the range of calculations around the parameter t.
breakpoints: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
plot: fixed by default to "FALSE". plot="FALSE" if a plot of the fuzzy number is not required.
Value

Returns a vector of decimals representing the bootstrap distribution of LR.

\[
\text{boot.mean.ml} 
\]

Estimates the bootstrap distribution of the likelihood ratio LR by the Algorithm 1 or 2 using the mean

Description

Estimates the bootstrap distribution of the likelihood ratio LR by the Algorithm 1 or 2 using the mean

Usage

\[
\text{Usage:} 
\]

\[
\text{boot.mean.ml(} 
\begin{align*}
\text{data.fuzzified}, & \\
\text{algorithm}, & \\
\text{distribution}, & \\
\text{sig}, & \\
\text{nsim} = 100, & \\
\text{mu} = \text{NA}, & \\
\text{sigma} = \text{NA}, & \\
\text{step} = 0.1, & \\
\text{margin} = c(5, 5), & \\
\text{breakpoints} = 100, & \\
\text{plot} = \text{TRUE} 
\end{align*} 
) 
\]

Arguments

\[
\begin{align*}
\text{data.fuzzified} & \text{ a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.} \\
\text{algorithm} & \text{ an algorithm chosen between "algo1" or "algo2".} \\
\text{distribution} & \text{ a distribution chosen between "normal", "poisson", "Student" or "Logistic".} \\
\text{sig} & \text{ a numerical value representing the significance level of the test.} \\
\text{nsim} & \text{ an integer giving the number of replications needed in the bootstrap procedure. It is set to 100 by default.} \\
\text{mu} & \text{ if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.} \\
\text{sigma} & \text{ if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA.} \\
\text{step} & \text{ a numerical value fixed to 0.1, defining the step of iterations on the interval \([t-5; t+5]\).} 
\end{align*} 
\]
margin  an optional numerical couple of values fixed to [5; 5], representing the range of calculations around the parameter t.

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

plot fixed by default to "FALSE". plot="FALSE" if a plot of the fuzzy number is not required.

Value

Returns a vector of decimals representing the bootstrap distribution of LR.

Examples

```r
mat <- matrix(c(1,2,2,2,2,1),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
PA11 <- c(1,2)
data.fuzzified <- FUZZ(mat,mi=1,si=1,PA=PA11)
emp.dist <- boot.mean.ml(data.fuzzified, algorithm = "algo1", distribution = "normal", sig = 0.05, nsim = 5, sigma = 1)
eta.boot <- quantile(emp.dist, probs = 95/100)
```

---

cube  

Cube a number

Description

Cube a number

Usage

cube(x)

Arguments

x  Number to be cubed

Value

The cube of the input
**D2**

Calculates a distance by the D2 between fuzzy numbers

**Description**

Calculates a distance by the D2 between fuzzy numbers

**Usage**

\[ \text{D2}(X, Y, \text{breakpoints} = 100) \]

**Arguments**

- **X**: a fuzzy number.
- **Y**: a fuzzy number.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

A numerical value.

---

**Defuzz.FANOVA**

Defuzzify the fuzzy sums of squares calculated by a FANOVA model by an exact calculation or an approximation

**Description**

Defuzzify the fuzzy sums of squares calculated by a FANOVA model by an exact calculation or an approximation

**Usage**

```r
Defuzz.FANOVA(  
  res,  
  distance.type = "DSGD",  
  i = 1,  
  j = 1,  
  theta = 1/3,  
  thetas = 1,  
  p = 2,  
  q = 0.5,  
  breakpoints = 100  
)
```
Delta.pq

Arguments

res a result of a call of the function FANOVA, where method = "distance".

distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

i parameter of the density function of the Beta distribution, fixed by default to i = 1.

j parameter of the density function of the Beta distribution, fixed by default to j = 1.

theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.

thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns a list of all the arguments of the function, the defuzzified total, treatment and residuals sums of squares, the decision made etc.

Delta.pq

Calculates a distance by the d_Delta.pq between fuzzy numbers

Description

Calculates a distance by the d_Delta.pq between fuzzy numbers

Usage

Delta.pq(X, Y, p, q, breakpoints = 100)
**Delta_jki**

**Arguments**

- \( x \) a fuzzy number.
- \( Y \) a fuzzy number.
- \( p \) a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( \text{Rho}_p \) and \( \text{Delta}_pq \).
- \( q \) a decimal value between 0 and 1, referring to the parameter of the metric \( \text{Delta}_pq \).
- \( \text{breakpoints} \) a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

A numerical value.

**Description**

Calculates the factor \( \text{Delta}_jki \)

**Usage**

\[
\text{Delta}_jki(x, i, K)
\]

**Arguments**

- \( x \) a dataset.
- \( i \) an observation index.
- \( K \) the total number of linguistics in a sub-item.

**Value**

The response matrix of binary values (0 or 1) related to the answers of a particular dataset for its corresponding sub-items.
distance  
Calculates a distance between fuzzy numbers

Description
Calculates a distance between fuzzy numbers

Usage
```r
distance(
  X,
  Y,
  type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)
```

Arguments
- **X**: a fuzzy number.
- **Y**: a fuzzy number.
- **type**: type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
- **i**: parameter of the density function of the Beta distribution, fixed by default to i = 1.
- **j**: parameter of the density function of the Beta distribution, fixed by default to j = 1.
- **theta**: a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_wabl/ldev/rdev.
- **thetas**: a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
- **p**: a positive integer such that $1 \leq p < \infty$, referring to the parameter of the Rho_p and Delta_pq.
- **q**: a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
**DSGD**

**Value**

A numerical value.

**Examples**

```r
X <- TrapezoidalFuzzyNumber(1, 2, 3, 4)
Y <- TrapezoidalFuzzyNumber(4, 5, 6, 7)
distance(X, Y, type = "DSGD.G")
distance(X, Y, type = "GSGD")
```

---

**DSGD**  
*Calculates a distance by the SGD between fuzzy numbers*

**Description**

Calculates a distance by the SGD between fuzzy numbers

**Usage**

```r
DSGD(X, Y, i = 1, j = 1, breakpoints = 100, theta = 1/3)
```

**Arguments**

- `X`: a fuzzy number.
- `Y`: a fuzzy number.
- `i`: parameter of the density function of the Beta distribution, fixed by default to `i = 1`.
- `j`: parameter of the density function of the Beta distribution, fixed by default to `j = 1`.
- `breakpoints`: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- `theta`: a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: `d_Bertoluzza`, `d_mid/spr` and `d_phi-wabl/ldev/rdev`.

**Value**

A numerical value.
FANOVA

Calculates a distance by the $d_{DSGD.G}$ between fuzzy numbers

Description

Calculates a distance by the $d_{DSGD.G}$ between fuzzy numbers

Usage

$$DSGD.G(X, Y, i = 1, j = 1, \text{thetas} = 1, \text{breakpoints} = 100)$$

Arguments

- **X**: a fuzzy number.
- **Y**: a fuzzy number.
- **i**: parameter of the density function of the Beta distribution, fixed by default to $i = 1$.
- **j**: parameter of the density function of the Beta distribution, fixed by default to $j = 1$.
- **thetas**: a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the $d_{theta}$ star and the $d_{GSGD}$ distances.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.

FANOVA

Computes a FANOVA model by a convenient metric, an exact calculation or an approximation

Description

Computes a FANOVA model by a convenient metric, an exact calculation or an approximation
Usage

FANOVA(
    formula,
    dataset,
    data.fuzzified,
    sig,
    method,
    distance.type = "DSGD",
    i = 1,
    j = 1,
    theta = 1/3,
    thetas = 1,
    p = 2,
    q = 0.5,
    breakpoints = 100,
    int.method = "int.simpson",
    plot = TRUE
)

Arguments

formula a description of the model to be fitted.
dataset the data frame containing all the variables of the model.
data.fuzzified the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
sig a numerical value representing the significance level of the test.
method the choices are the following: "distance", "exact", "approximation".
distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhopt", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \(d_{Bertoluzza}, d_{mid/spr}\) and \(d_{phi-wabl/ldev/rdev}\).
thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the \(d_{theta}\) star and the \(d_{GSGD}\) distances.
p a positive integer such that 1 \(\leq\) p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
breakpoints  a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

int.method  the method of numerical integration. It is set by default to the Simpson method, i.e. int.method = "int.simpson".

plot  fixed by default to "TRUE". plot = "FALSE" if a plot of the fuzzy number is not required.

Value

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

Examples

mat <- matrix(c(1,1,1,1,1,1,1,2,2,2,2,3,2,3,4,2,3,3,2,4), ncol = 2)
data <- data.frame(mat)
data$X1 <- factor(data$X1)
MF121 <- TrapezoidalFuzzyNumber(0,1,1.2)
MF122 <- TrapezoidalFuzzyNumber(1.8,1.9,2.2,2.8)
MF123 <- TrapezoidalFuzzyNumber(1.9,2.3,3.1,3.3)
MF124 <- TrapezoidalFuzzyNumber(3.1,3.4,4.1,4.2)
PA12 <- c(1,2,3,4)
data.fuzzified <- GFUZZ(data, 1, 2, PA12, "Identical")
formula = X2 ~ X1
res <- FANOVA(formula, dataset = data, method = "distance", data.fuzzified = data.fuzzified,
sig = 0.05, distance.type = "wabl")

---

FANOVA.approximation  Computes a FANOVA model by an approximation

Description

Computes a FANOVA model by an approximation

Usage

FANOVA.approximation(
  formula,  
  dataset,  
  data.fuzzified,  
  sig,  
  breakpoints = 100,  
  int.method = "int.simpson",  
  plot = TRUE
)

**Arguments**

- **formula**: a description of the model to be fitted.
- **dataset**: the data frame containing all the variables of the model.
- **data.fuzzified**: the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
- **sig**: a numerical value representing the significance level of the test.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- **int.method**: the method of numerical integration. It is set by default to the Simpson method, i.e. `int.method="int.simpson"`.
- **plot**: fixed by default to "TRUE". `plot="FALSE"` if a plot of the fuzzy number is not required.

**Value**

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

---

**FANOVA.distance**

*Computes a FANOVA model by a convenient metric*

**Description**

Computes a FANOVA model by a convenient metric

**Usage**

```r
FANOVA.distance(
  formula,
  dataset,
  data.fuzzified,
  sig,
  distance.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)
```
Arguments

**formula**
a description of the model to be fitted.

**dataset**
the data frame containing all the variables of the model.

**data.fuzzified**
the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.

**sig**
a numerical value representing the significance level of the test.

**distance.type**
type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

**i**
parameter of the density function of the Beta distribution, fixed by default to i = 1.

**j**
parameter of the density function of the Beta distribution, fixed by default to j = 1.

**theta**
a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.

**thetas**
a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

**p**
a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

**q**
a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.

**breakpoints**
a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

---

**FANOVA.exact**
*Computes a FANOVA model by an exact calculation*

**Description**

Computes a FANOVA model by an exact calculation
FANOVA.summary

Usage

FANOVA.exact(  
    formula,  
    dataset,  
    data.fuzzified,  
    sig,  
    breakpoints = 100,  
    int.method = "int.simpson",  
    plot = TRUE
)

Arguments

formula a description of the model to be fitted.
dataset the data frame containing all the variables of the model.
data.fuzzified the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
sig a numerical value representing the significance level of the test.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
int.method the method of numerical integration. It is set by default to the Simpson method, i.e. int.method="int.simpson".
plot fixed by default to "TRUE", plot="FALSE" if a plot of the fuzzy number is not required.

Value

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

FANOVA.summary  Prints the summary of the estimation of a FANOVA metric-based model

Description

Prints the summary of the estimation of a FANOVA metric-based model

Usage

FANOVA.summary(res)

Arguments

res a result of a call of the function FANOVA, where method = "distance".
Value

Returns a list of summary statistics of the estimated model given in res, shown in a FANOVA table. In addition, the F-statistics with their p-values, and the decision are given.

---

**fci.ml**

*Estimates a fuzzy confidence interval by the Likelihood method*

Description

Estimates a fuzzy confidence interval by the Likelihood method

Usage

```
fci.ml(
  data.fuzzified, 
  t, 
  distribution, 
  sig, 
  mu = NA, 
  sigma = NA, 
  step = 0.05, 
  margin = c(5, 5), 
  breakpoints = 100, 
  plot = TRUE
)
```

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
t a given numerical or fuzzy type parameter of the distribution.
distribution a distribution chosen between "normal", "poisson", "Student" or "Logistic".
sig a numerical value representing the significance level of the test.
mu if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.
sigma if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA.
step a numerical value fixed to 0.05, defining the step of iterations on the interval [t-5; t+5].
margin an optional numerical couple of values fixed to [5; 5], representing the range of calculations around the parameter t.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
plot fixed by default to "FALSE". plot="FALSE" if a plot of the fuzzy number is not required.
Value

Returns a matrix composed by 2 vectors representing the numerical left and right alpha-cuts. For this output, is.alphacuts = TRUE.

Examples

data <- matrix(c(1,2,3,2,1,1,3,1,2),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF113 <- TrapezoidalFuzzyNumber(2,3,3,4)
PA11 <- c(1,2,3)
data.fuzzified <- FUZZ(data,mi=1,si=1,PA=PA11)
Fmean <- Fuzzy.sample.mean(data.fuzzified)
fci.ml(data.fuzzified, t = Fmean, distribution = "normal", sig= 0.05, sigma = 0.62)

fci.ml.boot

Estimates a fuzzy confidence interval by the Likelihood method

Description

Estimates a fuzzy confidence interval by the Likelihood method

Usage

fci.ml.boot(
data.fuzzified, 
t, 
distribution, 
sig, 
coef.boot, 
mu = NA, 
sigma = NA, 
step = 0.05, 
margin = c(5, 5), 
breakpoints = 100, 
plot = TRUE
)

Arguments

data.fuzzified  a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
t  a given numerical or fuzzy type parameter of the distribution.
distribution  a distribution chosen between "normal", "poisson", "Student" or "Logistic".
sig  a numerical value representing the significance level of the test.
coef.boot  a decimal representing the 1-sig-quantile of the bootstrap distribution of LR.
mu if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.

sigma if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA.

step a numerical value fixed to 0.05, defining the step of iterations on the interval [t-5; t+5].

margin an optional numerical couple of values fixed to [5; 5], representing the range of calculations around the parameter t.

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

plot fixed by default to "FALSE", plot="FALSE" if a plot of the fuzzy number is not required.

Value

Returns a matrix composed by 2 vectors representing the numerical left and right alpha-cuts. For this output, is.alphacuts = TRUE.

Examples

data <- matrix(c(1,2,3,2,1,3,1,2),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF113 <- TrapezoidalFuzzyNumber(2,3,3,4)
PA11 <- c(1,2,3)
data.fuzzified <- FUZZ(data,mi=1,si=1,PA=PA11)
Fmean <- Fuzzy.sample.mean(data.fuzzified)
fc1.ml.boot(data.fuzzified, t = Fmean, distribution = "normal", sig= 0.05, sigma = 0.62, coef.boot = 1.8225)

FMANOVA

Computes a Mult-FANOVA model by a convenient metric, an exact calculation or an approximation

Description

Computes a Mult-FANOVA model by a convenient metric, an exact calculation or an approximation

Usage

FMANOVA(
  formula, 
  dataset, 
  data.fuzzified, 
  sig = 0.05, 
  method, 
  distance.type = "DSGD",
)
index.var = NA,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100,
  int.method = "int.simpson",
  plot = TRUE
)

Arguments

formula  a description of the model to be fitted.
dataset  the data frame containing all the variables of the model.
data.fuzzified the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
sig a numerical value representing the significance level of the test.
method the choices are the following: "distance", "exact", "approximation".
distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
index.var the column index of the considered variable for which the output will be printed. It is an argument of the Mult-FANOVA models by the exact and the approximation methods only.
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
int.method the method of numerical integration. It is set by default to the Simpson method, i.e. int.method="int.simpson".
plot = FALSE if a plot of the fuzzy number is not required.

Value

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

Examples

```r
mat <- matrix(c(2,2,1,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,1,1,1,1,2,3,4,3,1,2,5,4,4,3),ncol=3)
data <- data.frame(mat)
MF131 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF132 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF133 <- TrapezoidalFuzzyNumber(2,3,3,4)
MF134 <- TrapezoidalFuzzyNumber(3,4,4,5)
MF135 <- TrapezoidalFuzzyNumber(4,5,5,6)
PA13 <- c(1,2,3,4,5); mi <- 1; si <- 3
Yfuzz <- FUZZ(data,1,3,PA13)
attach(data)
formula <- X3 ~ X1 + X2
res <- FMANOVA(formula, data, Yfuzz, method = "distance", distance.type = "wabl")
detach(data)
```

FMANOVA.approximation

Computes a Mult-FANOVA model by an approximation

Description

Computes a Mult-FANOVA model by an approximation

Usage

```r
FMANOVA.approximation(
  formula,
  dataset,
  data.fuzzified,
  sig = 0.05,
  breakpoints = 100,
  index.var = NA,
  int.method = "int.simpson",
  plot = TRUE
)
```
FMANOVA.distance

Arguments

- **formula**: a description of the model to be fitted.
- **dataset**: the data frame containing all the variables of the model.
- **data.fuzzified**: the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
- **sig**: a numerical value representing the significance level of the test.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- **index.var**: the column index of the considered variable for which the output will be printed. It is an argument of the Mult-FANOVA models by the exact and the approximation methods only.
- **int.method**: the method of numerical integration. It is set by default to the Simpson method, i.e. int.method="int.simpson".
- **plot**: fixed by default to "TRUE". plot="FALSE" if a plot of the fuzzy number is not required.

Value

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

FMANOVA.distance  Computes a Mult-FANOVA model by a convenient metric

Description

Computes a Mult-FANOVA model by a convenient metric

Usage

```r
FMANOVA.distance(
    formula,
    dataset,
    data.fuzzified,
    distance.type,
    sig = 0.05,
    i = 1,
    j = 1,
    theta = 1/3,
    thetas = 1,
    p = 2,
    q = 0.5,
    breakpoints = 100
)
```
Arguments

formula a description of the model to be fitted.
dataset the data frame containing all the variables of the model.
data.fuzzified the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
sig a numerical value representing the significance level of the test.
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
theetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

FMANOVA.exact Computes a Mult-FANOVA model by an exact calculation

Description

Computes a Mult-FANOVA model by an exact calculation
**FMANOVA.interaction.summary**

**Usage**

```
FMANOVA.exact(
    formula,
    dataset,
    data.fuzzified,
    sig = 0.05,
    breakpoints = 100,
    int.method = "int.simpson",
    index.var = NA,
    plot = TRUE
)
```

**Arguments**

- `formula`: a description of the model to be fitted.
- `dataset`: the data frame containing all the variables of the model.
- `data.fuzzified`: the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
- `sig`: a numerical value representing the significance level of the test.
- `breakpoints`: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- `int.method`: the method of numerical integration. It is set by default to the Simpson method, i.e. `int.method="int.simpson"`.
- `index.var`: the column index of the considered variable for which the output will be printed. It is an argument of the Mult-FANOVA models by the exact and the approximation methods only.
- `plot`: fixed by default to "TRUE". plot="FALSE" if a plot of the fuzzy number is not required.

**Value**

Returns a list of all the arguments of the function, the total, treatment and residuals sums of squares, the coefficients of the model, the test statistics with the corresponding p-values, and the decision made.

---

**FMANOVA.interaction.summary**

Prints the summary of the estimation of the interaction in a Mult-FANOVA metric-based model

**Description**

Prints the summary of the estimation of the interaction in a Mult-FANOVA metric-based model
Usage

FMANOVA.interaction.summary(res)

Arguments

res a result of a call of the function FMANOVA, where method = "distance".

Value

Returns a list of summary statistics of the estimated model given in res, shown in a FANOVA table. In addition, the F-statistics with their p-values, and the decision are given.

Description

Prints the summary of the estimation of a Mult-FANOVA metric-based model

Usage

FMANOVA.summary(res)

Arguments

res a result of a call of the function FMANOVA, where method = "distance".

Value

Returns a list of summary statistics of the estimated model given in res, shown in a FANOVA table. In addition, the F-statistics with their p-values, and the decision are given.

Description

Calculates multiple tests corresponding to the fuzzy response variable

Usage

Ftests(test)

Arguments

test a result of a call of the function FMANOVA.
FTukeyHSD

Value

Returns a table of the following different indicators "Wilks","F-Wilks", "Hotelling-Lawley trace" and "Pillai Trace".

Examples

```r
mat <- matrix(c(2,2,1,2,2,2,2,2,2,2,1,1,1,2,3,4,3,1,2,5,4,3),ncol=3)
data <- data.frame(mat)
MF131 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF132 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF133 <- TrapezoidalFuzzyNumber(2,3,3,4)
MF134 <- TrapezoidalFuzzyNumber(3,4,4,5)
MF135 <- TrapezoidalFuzzyNumber(4,5,5,6)
PA13 <- c(1,2,3,4,5); mi <- 1; si <- 3
Yfuzz <- FUZZ(data,1,3,PA13)
attach(data)
formula <- X3 ~ X1 + X2
res <- FMANOVA(formula, data, Yfuzz, method = "distance", distance.type = "wabl")
Ftests(res)
detach(data)
```

FTukeyHSD

Calculates the Tukey HSD test corresponding to the fuzzy response variable

Description

Calculates the Tukey HSD test corresponding to the fuzzy response variable

Usage

```
FTukeyHSD(test, variable, cont = c(1, -1), conf.level = 0.95)
```

Arguments

- `test`: a result of a call of the function FMANOVA.
- `variable`: the name of a variable in the data set.
- `cont`: the contrasts of the model. It is set by default to c(1,-1).
- `conf.level`: the confidence level of the test. It is set by default to 0.95.

Value

Returns a table of comparisons of means of the different levels of a given factor, two by two. The table contains the means of populations, the lower and upper bounds of the confidence intervals, and their p-values.
Examples

```r
mat <- matrix(c(2,2,1,2,2,2,2,2,2,2,2,1,1,1,1,2,3,4,4,3,1,2,5,4,4,3),ncol=3)
data <- data.frame(mat)
MF131 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF132 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF133 <- TrapezoidalFuzzyNumber(2,3,3,4)
MF134 <- TrapezoidalFuzzyNumber(3,4,4,5)
MF135 <- TrapezoidalFuzzyNumber(4,5,5,6)
PA13 <- c(1,2,3,4,5); mi <- 1; si <- 3
Yfuzz <- FUZZ(data,1,3,PA13)
attach(data)
formula <- X3 ~ X1 + X2
res <- FMANOVA(formula, data, Yfuzz, method = "distance", distance.type = "wabl")
FTukeyHSD(res, "X1")[[1]]
detach(data)
```

FUZZ

Fuzzifies a variable modelled by trapezoidal or triangular fuzzy numbers

Description

Fuzzifies a variable modelled by trapezoidal or triangular fuzzy numbers

Usage

```r
FUZZ(data, mi, si, PA)
```

Arguments

- **data**: a data set.
- **mi**: the index of the main-item containing the concerned variable.
- **si**: the index of the sub-item of a given main-item mi.
- **PA**: a vector of the linguistic terms of the considered variable.

Value

A fuzzification matrix composed by 4 columns c(p,q,r,s), and m lines, i.e. number of observations. No NA is allowed.

Examples

```r
data <- matrix(c(1,2,3,2,2,1,3,1,2),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF113 <- TrapezoidalFuzzyNumber(2,3,3,3)
PA11 <- c(1,2,3)
data.fuzzified <- FUZZ(data,mi=1,si=1,PA=PA11)
is.trfuzzification(data.fuzzified)
```
**Fuzzy.CI.ML.test**

*Computes a fuzzy inference test by the fuzzy confidence intervals method calculated by the Likelihood method*

---

**Description**

Computes a fuzzy inference test by the fuzzy confidence intervals method calculated by the Likelihood method.

**Usage**

```r
Fuzzy.CI.ML.test(
  data.fuzzified,
  H0,
  H1,
  t,
  mu = NA,
  sigma = NA,
  sig,
  distribution,
  coef.boot,
  distance.type = "DSGD",
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100,
  step = 0.05,
  margin = c(5, 5),
  plot = TRUE
)
```

**Arguments**

- `data.fuzzified`: a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
- `H0`: a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
- `H1`: a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
- `t`: a given numerical or fuzzy type parameter of the distribution.
- `mu`: if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.
if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA.

a numerical value representing the significance level of the test.

a distribution chosen between "normal", "poisson", "Student" or "Logistic".

a decimal representing the 1-sig-quantile of the bootstrap distribution of LR.

type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhoph", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

parameter of the density function of the Beta distribution, fixed by default to i = 1.

parameter of the density function of the Beta distribution, fixed by default to j = 1.

a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{Bertoluzza}, d_{mid/spr} \) and \( d_{phi-wabl/ldev/rdev} \).

a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the \( d_{theta} \) star and the \( d_{GSGD} \) distances.

a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( \text{Rho}_p \) and \( \text{Delta}_{pq} \). By default, p is fixed to 2.

a decimal value between 0 and 1, referring to the parameter of the metric \( \text{Delta}_{pq} \). By default, p is fixed to 0.5.

a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

a numerical value fixed to 0.05, defining the step of iterations on the interval \([t-5; t+5]\).

an optional numerical couple of values fixed to \([5; 5]\), representing the range of calculations around the parameter t.

fixed by default to "FALSE". plot="FALSE" if a plot of the fuzzy number is not required.

Returns a list composed by the arguments, the fuzzy confidence intervals, the fuzzy decisions, the defuzzified values and the decision made.

Examples

```r
data <- matrix(c(1,2,3,2,1,3,1,2),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF113 <- TrapezoidalFuzzyNumber(2,3,3,4)
PA11 <- c(1,2,3)
data.fuzzified <- FUZZ(data,mi=1,si=1,PA=PA11)
```
Fuzzy.CI.test <- Fuzzy.sample.mean(data.fuzzified)
H0 <- TriangularFuzzyNumber(2.2,2.5,3)
H1 <- TriangularFuzzyNumber(2.5,2.5,5)
coef.boot <- 3.494829
(res <- Fuzzy.CI.ML.test(data.fuzzified, H0, H1, t = Fmean, sigma=0.7888, coef.boot = coef.boot, sig=0.05, distribution="normal", distance.type="GSGD"))
res$decision

Fuzzy.CI.test

Computes a fuzzy inference test by the traditional fuzzy confidence intervals

Description

Computes a fuzzy inference test by the traditional fuzzy confidence intervals

Usage

Fuzzy.CI.test(
  type,  # a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
  H0,  # a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
  H1,  # a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
  t,    # a given numerical or fuzzy type parameter of the distribution.
  s.d,  #
  n,    #
  sig,  #
  distribution,  #
  distance.type = "DSGD",  #
  i = 1,  #
  j = 1,  #
  theta = 1/3,  #
  thetas = 1,  #
  p = 2,  #
  q = 0.5,  #
  breakpoints = 100,  #
  plot = TRUE  #
)

Arguments

- **type** a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
- **H0** a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
- **H1** a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
- **t** a given numerical or fuzzy type parameter of the distribution.
Fuzzy.CI.test

s.d a numerical value for the standard deviation of the distribution.
n the total number of observations of the data set.
sig a numerical value representing the significance level of the test.
distribution a distribution chosen between "normal", "poisson", "Student" or "Logistic".
distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhopt", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to \( i = 1 \).
j parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).
theta a numerical value between 0 and 1, representing a weighting parameter. By default, \( \theta \) is fixed to \( \frac{1}{3} \) referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{\text{Bertoluzza}} \), \( d_{\text{mid/spr}} \) and \( d_{\text{phi-wabl/ldev/rdev}} \).
thesas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, \( \text{thesas} \) is fixed to 1. This parameter is used in the calculations of the \( d_{\text{theta star}} \) and the \( d_{\text{GSGD}} \) distances.
p a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( \text{Rho}_p \) and \( \text{Delta}_pq \). By default, \( p \) is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric \( \text{Delta}_pq \). By default, \( p \) is fixed to 0.5.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
plot a logical rule "TRUE" or "FALSE" for defining whether to plot the corresponding graphs or not.

Value

Returns a list composed by the arguments, the fuzzy confidence intervals, the fuzzy decisions, the defuzzified values and the decision made.

Examples

```r
H0 <- TriangularFuzzyNumber(2.9,3,3.1)
H1 <- TriangularFuzzyNumber(3,3,5)
res <- Fuzzy.CI.test(type = 0, H0, H1, t = TriangularFuzzyNumber(0.8,1.80,2.80), s.d = 0.79, n = 10, sig = 0.05, distribution = "normal", distance.type="GSGD")
```
Fuzzy.decisions

Computes the fuzzy decisions of a fuzzy inference test by the traditional fuzzy confidence intervals

Description

Computes the fuzzy decisions of a fuzzy inference test by the traditional fuzzy confidence intervals

Usage

Fuzzy.decisions(
  type,
  H0,  # a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
  H1,  # a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
  t,  # a given numerical or fuzzy type parameter of the distribution.
  s.d,  # a numerical value for the standard deviation of the distribution.
  n,  # the total number of observations of the data set.
  sig,  # a numerical value representing the significance level of the test.
  distribution,  # a distribution chosen between "normal", "poisson", "Student" or "Logistic".
  distance.type = "DSGD",  # type of distance chosen from the family of distances. The different choices
  i = 1,  # are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr",
  j = 1,  # "wabl", "DSGD", "DSGD.G", "GSGD".
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)

Arguments

type a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
H0 a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
H1 a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
t a given numerical or fuzzy type parameter of the distribution.
s.d a numerical value for the standard deviation of the distribution.
n the total number of observations of the data set.
sig a numerical value representing the significance level of the test.
distribution a distribution chosen between "normal", "poisson", "Student" or "Logistic".
distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i  parameter of the density function of the Beta distribution, fixed by default to \( i = 1 \).

j  parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).

\( \theta \)  a numerical value between 0 and 1, representing a weighting parameter. By default, \( \theta \) is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{\text{Bertoluzza}} \), \( d_{\text{mid/spr}} \) and \( d_{\text{phi-wabl/ldev/rdev}} \).

\( \theta_\text{tas} \)  a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, \( \theta_\text{tas} \) is fixed to 1. This parameter is used in the calculations of the \( d_{\theta \text{ta}} \) and the \( d_{\text{GSGD}} \) distances.

\( p \)  a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( \rho_p \) and \( \Delta_{pq} \). By default, \( p \) is fixed to 2.

\( q \)  a decimal value between 0 and 1, referring to the parameter of the metric \( \Delta_{pq} \). By default, \( p \) is fixed to 0.5.

breakpoints  a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

Returns a list composed by the arguments, the fuzzy confidence intervals and their complements, the fuzzy decisions and the defuzzified values.

**Examples**

```r
H0 <- alphacut(TriangularFuzzyNumber(2.9, 3, 3.1), seq(0, 1, 0.01))
H1 <- alphacut(TriangularFuzzyNumber(3, 3, 5), seq(0, 1, 0.01))
t <- alphacut(TriangularFuzzyNumber(0.8, 1.80, 2.80), seq(0, 1, 0.01))
res <- Fuzzy.decisions(type = 0, H0, H1, t = t, s.d = 0.79, n = 10, sig = 0.05, distribution = "normal", distance.type = "GSGD")
```

**Description**

Computes the fuzzy decisions of a fuzzy inference test by the fuzzy confidence intervals by the likelihood method.

**Usage**

```r
Fuzzy.decisions.ML(
  data.fuzzified,
  H0,
  H1,
```
t, coef.boot, mu = NA, sigma = NA, sig, distribution, distance.type = "DSGD", i = 1, j = 1, theta = 1/3, thetas = 1, p = 2, q = 0.5, breakpoints = 100, step = 0.05, margin = c(5, 5), plot = FALSE
)

Arguments

data.fuzzified: a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

H0: a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.

H1: a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.

t: a given numerical or fuzzy type parameter of the distribution.

coef.boot: a decimal representing the 1-sig-quantile of the bootstrap distribution of LR.

mu: if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.

sigma: if the standard deviation of the normal distribution is known, sigma should be a numerical value. Otherwise, the argument sigma is fixed to NA.

sig: a numerical value representing the significance level of the test.

distribution: a distribution chosen between "normal", "poisson", "Student" or "Logistic".

distance.type: type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

i: parameter of the density function of the Beta distribution, fixed by default to i = 1.

j: parameter of the density function of the Beta distribution, fixed by default to j = 1.

theta: a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
Fuzzy.Difference

Fuzzy.Difference

Calculates the difference between two fuzzy numbers

Description

Calculates the difference between two fuzzy numbers

Usage

Fuzzy.Difference(X, Y, alphacuts = FALSE, breakpoints = 100)
Arguments

X  a fuzzy number of any type.
Y  a fuzzy number of any type.
alphacuts  fixed by default to "FALSE". No alpha-cuts are printed in this case.
breakpoints  a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

If the parameter alphacuts="TRUE", the function returns a matrix composed by 2 vectors representing the left and right alpha-cuts. For this output, is.alphacuts = TRUE. If the parameter alphacuts="FALSE", the function returns a trapezoidal fuzzy number given by the quadruple (p,q,r,s), such that p ≤ q ≤ r ≤ s.

Examples

X <- TrapezoidalFuzzyNumber(5,6,7,8)
Y <- TrapezoidalFuzzyNumber(1,2,3,4)
Fuzzy.Difference(X,Y)

Fuzzy.exact.variance  Calculates the exact variance

Description

Calculates the exact variance

Usage

Fuzzy.exact.variance(data.fuzzified, breakpoints = 100, plot = FALSE)

Arguments

data.fuzzified  a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
breakpoints  a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
plot  fixed by default to "FALSE". plot="TRUE" if a plot of the fuzzy number is required.

Value

The numerical alpha-cuts of the estimated fuzzy variance.
Fuzzy.exact.variance.poly.left

Gives the polynomial forms of the numerical alpha-cuts modelling the exact variance

Description

Gives the polynomial forms of the numerical alpha-cuts modelling the exact variance

Usage

Fuzzy.exact.variance.poly.left(data.fuzzified, breakpoints = 100)

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A table composed by the coefficients of the second order equations of the left side, given at the corresponding definitions domains.

Fuzzy.exact.variance.poly.right

Gives the polynomial forms of the numerical alpha-cuts modelling the exact variance

Description

Gives the polynomial forms of the numerical alpha-cuts modelling the exact variance

Usage

Fuzzy.exact.variance.poly.right(data.fuzzified, breakpoints = 100)

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
**Fuzzy.p.value**

Value
A table composed by the coefficients of the second order equations of the right side, given at the corresponding definitions domains.

---

Fuzzy.p.value  **Computes the fuzzy p-value of a given fuzzy hypothesis test**

**Description**
Computes the fuzzy p-value of a given fuzzy hypothesis test

**Usage**

```r
Fuzzy.p.value(  
  type,  
  H0,  
  H1,  
  t,  
  s.d = 1,  
  n,  
  sig,  
  distribution,  
  distance.type = "DSGD",  
  i = 1,  
  j = 1,  
  theta = 1/3,  
  thetas = 1,  
  p = 2,  
  q = 0.5,  
  breakpoints = 100  
)
```

**Arguments**

- **type**
  a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.

- **H0**
  a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.

- **H1**
  a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.

- **t**
  a given numerical or fuzzy type parameter of the distribution.

- **s.d**
  a numerical value for the standard deviation of the distribution.

- **n**
  the total number of observations of the data set.

- **sig**
  a numerical value representing the significance level of the test.
distribution a distribution chosen between "normal", "poisson", "Student" or "Logistic".
distance.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
brbreakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns the defuzzified p-value and the decision made.

Examples

```r
H0 <- TriangularFuzzyNumber(2.2, 2.5, 3)
H1 <- TriangularFuzzyNumber(2.5, 2.5, 5)
Fuzzy.p.value(type=1, H0, H1, t=TriangularFuzzyNumber(0.8, 1.8, 2.8),
 s.d=0.7888, n=10, sig=0.05, distribution="normal", distance.type="GSGD")
```

---

Fuzzy.p.value.mean Computes the fuzzy p-value of a given fuzzy hypothesis test for the mean

Description

Computes the fuzzy p-value of a given fuzzy hypothesis test for the mean
**Usage**

```r
Fuzzy.p.value.mean(
  data.fuzzified,
  type,
  H0,
  H1,
  s.d = 1,
  sig,
  distribution,
  distance.type = "DSGD",
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)
```

**Arguments**

- `data.fuzzified`: a fuzzification matrix constructed by a call to the function `FUZZ` or the function `GFUZZ`, or a similar matrix. No NA are allowed.
- `type`: a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
- `H0`: a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
- `H1`: a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
- `s.d`: a numerical value for the standard deviation of the distribution.
- `sig`: a numerical value representing the significance level of the test.
- `distribution`: a distribution chosen between "normal", "poisson" or "Student".
- `distance.type`: type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
- `i`: parameter of the density function of the Beta distribution, fixed by default to \( i = 1 \).
- `j`: parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).
- `theta`: a numerical value between 0 and 1, representing a weighting parameter. By default, \( \theta \) is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{Bertoluzza} \), \( d_{mid/spr} \) and \( d_{phi-wabl/ldev/rdev} \).
- `thetas`: a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, \( \theta \) is fixed to 1. This parameter is used in the calculations of the \( d_{\theta} \) star and the \( d_{GSGD} \) distances.
p  
a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

q  
a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.

breakpoints  
a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns the defuzzified p-value and the decision made.

Examples

data <- matrix(c(1,2,3,2,2,1,1,3,1,2),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF113 <- TrapezoidalFuzzyNumber(2,3,3,4)
PA11 <- c(1,2,3)
data.fuzzified <- FUZZ(data,mi=1,si=1,PA=PA11)
H0 <- TriangularFuzzyNumber(2.2,2.5,3)
H1 <- TriangularFuzzyNumber(2.5,2.5,5)
Fuzzy.p.value.mean(data.fuzzified, type=1, H0, H1, s.d=0.7888, sig=0.05, distribution="normal", distance.type="GSGD")

fuzzy.predicted.values

Calculates the fuzzy predicted values

Description

Calculates the fuzzy predicted values

Usage

fuzzy.predicted.values(dataset, coef.model)

Arguments

dataset  
the data frame containing all the variables of the model.

coef.model  
the coefficients of the model.

Value

Returns a matrix containing the alpha-cuts of the fuzzy predicted values.
### fuzzy.residuals

*Calculates the fuzzy residuals*

**Description**

Calculates the fuzzy residuals

**Usage**

```r
fuzzy.residuals(data.fuzzified, predicted.values)
```

**Arguments**

- `data.fuzzified`: the fuzzified data set constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.
- `predicted.values`: the fuzzy predicted values constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix.

**Value**

Returns a matrix containing the alpha-cuts of the fuzzy residuals.

---

### Fuzzy.sample.mean

*Calculates the fuzzy sample mean*

**Description**

Calculates the fuzzy sample mean

**Usage**

```r
Fuzzy.sample.mean(data.fuzzified, breakpoints = 100, alphacuts = FALSE)
```

**Arguments**

- `data.fuzzified`: a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
- `breakpoints`: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- `alphacuts`: fixed by default to "FALSE". No alpha-cuts are printed in this case.
If the parameter alphacuts="TRUE", the function returns a matrix composed by 2 vectors representing the numerical left and right alpha-cuts. For this output, is.alphacuts = TRUE. If the parameter alphacuts="FALSE", the function returns a trapezoidal fuzzy number given by the quadruple (p,q,r,s).

Examples

mat <- matrix(c(1,2,2,3,3,4,4,5), ncol =4)
Fuzzy.sample.mean(mat)

Fuzzy.sample.variance.approximation

Fuzzy sample variance (approx) - general

Description

Fuzzy sample variance (approx) - general

Usage

Fuzzy.sample.variance.approximation(data.fuzzified, appro.id)

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
appro.id an integer between 1 and 5 giving the method of approximation chosen.

Value

A numerical value.

Fuzzy.sample.variance.approximation1

Fuzzy sample variance (approx) - method 1

Description

Fuzzy sample variance (approx) - method 1

Usage

Fuzzy.sample.variance.approximation1(data.fuzzified)
Fuzzy.sample.variance.approximation2

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

Value

A numerical value.

Fuzzy.sample.variance.approximation3

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
Fuzzy sample variance (approx) - method 4

Description
Fuzzy sample variance (approx) - method 4

Usage
Fuzzy.sample.variance.approximation4(data.fuzzified)

Arguments
data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

Value
A numerical value.

Fuzzy sample variance (approx) - method 5

Description
Fuzzy sample variance (approx) - method 5

Usage
Fuzzy.sample.variance.approximation5(data.fuzzified)

Arguments
data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

Value
A numerical value.
**Fuzzy.Square**

*Calculates numerically the square of a fuzzy number*

**Description**
Calculates numerically the square of a fuzzy number

**Usage**

Fuzzy.Square(F1L, breakpoints = 100, plot = FALSE)

**Arguments**
- **F1L**: a fuzzy number.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- **plot**: fixed by default to "FALSE". plot="TRUE" if a plot of the fuzzy number is required.

**Value**
A matrix composed by 2 vectors representing the numerical left and right alpha-cuts. For this output, is.alphacuts = TRUE.

**Examples**

```r
X <- TrapezoidalFuzzyNumber(1,2,3,4)
Fuzzy.Square(X, plot=TRUE)
```

---

**Fuzzy.Square.poly.left**

*Gives the polynomial expression of the left alpha-levels of the numerical square of a fuzzy number*

**Description**
Gives the polynomial expression of the left alpha-levels of the numerical square of a fuzzy number

**Usage**

Fuzzy.Square.poly.left(F1L, breakpoints = 100)

**Arguments**
- **F1L**: a fuzzy number.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
Value
A table containing print the related polynoms at the corresponding definition domains.

Examples

X <- TrapezoidalFuzzyNumber(1,2,3,4)
Fuzzy.Square.poly.left(X)

Description
Gives the polynomial expression of the right alpha-levels of the numerical square of a fuzzy number.

Usage

Fuzzy.Square.poly.right(F1L, breakpoints = 100)

Arguments

F1L a fuzzy number.

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value
A table containing print the related polynoms at the corresponding definition domains.

Examples

X <- TrapezoidalFuzzyNumber(1,2,3,4)
Fuzzy.Square.poly.right(X)
Fuzzy.variance calculates the variance by a chosen method: distance, exact or approximation.

**Description**

Calculates the variance by a chosen method: distance, exact or approximation.

**Usage**

```r
Fuzzy.variance(
  data.fuzzified,
  method,
  dist.type = "DSGD",
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100,
  int.method = "int.simpson",
  plot = FALSE
)
```

**Arguments**

- `data.fuzzified`: a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
- `method`: choices are the following: "distance", "exact", "approximation1", "approximation2", "approximation3", "approximation4", "approximation5".
- `dist.type`: type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
- `i`: parameter of the density function of the Beta distribution, fixed by default to i = 1.
- `j`: parameter of the density function of the Beta distribution, fixed by default to j = 1.
- `theta`: a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
- `thetas`: a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
GaussianBellFuzzyNumber

Description

Creates a Gaussian two-sided bell fuzzy number

Usage

GaussianBellFuzzyNumber(
  left.mean,
  left.sigma,
  right.mean,
  right.sigma,
  alphacuts = FALSE,
  margin = c(5, 5),
  step = 0.01,
  breakpoints = 100,
  precision = 4,
  plot = FALSE
)
GaussianFuzzyNumber

Arguments

left.mean  a numerical value of the parameter mu of the left Gaussian curve.
left.sigma a numerical value of the parameter sigma of the left Gaussian curve.
right.mean a numerical value of the parameter mu of the right Gaussian curve.
right.sigma a numerical value of the parameter sigma of the right Gaussian curve.
alphacuts fixed by default to "FALSE". No alpha-cuts are printed in this case.
margin an optional numerical couple of values representing the range of calculations of
        the Gaussian curve written as [mean - 3*sigma; mean + 3*sigma] by default.
step a numerical value fixing the step between two knots dividing the interval [mean
        - 3*sigma; mean + 3*sigma].
breakpoints a positive arbitrary integer representing the number of breaks chosen to build
          the numerical alpha-cuts. It is fixed to 100 by default.
precision an integer specifying the number of decimals for which the calculations are
          made. These latter are set by default to be at the order of 1/10^4.
plot fixed by default to "FALSE". plot="TRUE" if a plot of the fuzzy number is
        required.

Value

If the parameter alphacuts="TRUE", the function returns a matrix composed by 2 vectors repre-
senting the left and right alpha-cuts. For this output, is.alphacuts = TRUE. If the parameter alphacuts="FALSE", the function returns a list composed by the Class, the mean, the sigma, the vectors
of the left and right alpha-cuts.

Examples

GBFN <- GaussianBellFuzzyNumber(left.mean = -1, left.sigma = 1,
right.mean = 2, right.sigma = 1, alphacuts = TRUE, plot=TRUE)
is.alphacuts(GBFN)

GaussianFuzzyNumber  Creates a Gaussian fuzzy number

Description

Creates a Gaussian fuzzy number

Usage

GaussianFuzzyNumber(
  mean,
  sigma,
  alphacuts = FALSE,
  margin = c(5, 5),
)
step = 0.01,
brbreakpoints = 100,
precision = 4,
plot = FALSE
)

Arguments

- **mean**: a numerical value of the parameter mu of the Gaussian curve.
- **sigma**: a numerical value of the parameter sigma of the Gaussian curve.
- **alphacuts**: fixed by default to "FALSE". No alpha-cuts are printed in this case.
- **margin**: an optional numerical couple of values representing the range of calculations of the Gaussian curve written as [mean - 3*sigma; mean + 3*sigma] by default.
- **step**: a numerical value fixing the step between two knots dividing the interval [mean - 3*sigma; mean + 3*sigma].
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
- **precision**: an integer specifying the number of decimals for which the calculations are made. These latter are set by default to be at the order of 1/10^4.
- **plot**: fixed by default to "FALSE". plot="TRUE" if a plot of the fuzzy number is required.

Value

If the parameter alphacuts="TRUE", the function returns a matrix composed by 2 vectors representing the left and right alpha-cuts. For this output, is.alphacuts = TRUE. If the parameter alphacuts="FALSE", the function returns a list composed by the Class, the mean, the sigma, the vectors of the left and right alpha-cuts.

Examples

```r
GFN <- GaussianFuzzyNumber(mean = 0, sigma = 1, alphacuts = TRUE, plot=TRUE)
is.alphacuts(GFN)
```

---

**GFUZZ**

Fuzzifies a variable modelled by any type of fuzzy numbers

Description

Fuzzifies a variable modelled by any type of fuzzy numbers

Usage

```r
GFUZZ(data, mi, si, PA, spec = "Identical", breakpoints = 100)
```
GLOB.EVAL

Calculates the global evaluation of a linguistic questionnaire

Arguments

data a data set.
mi the index of the main-item containing the concerned variable.
si the index of the sub-item of a given main-item mi.
PA a vector of the linguistic terms of the considered variable.
spec specification of the fuzzification matrix. The possible values are "Identical" and "Not Identical".
brackets a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. breakpoints is fixed to 100 by default.

Value

A numerical fuzzification array of 3 dimensions (m,n,2), with m lines, n columns and no NA.

Examples

data <- matrix(c(1,2,3,2,1,3,1,2),ncol=1)
MF111 <- TrapezoidalFuzzyNumber(0,1,1,2)
MF112 <- TrapezoidalFuzzyNumber(1,2,2,3)
MF113 <- TrapezoidalFuzzyNumber(2,3,3,3)
PA11 <- c(1,2,3)
data.fuzzified <- GFUZZ(data,mi=1,si=1,PA=PA11)

Description

Calculates the global evaluation of a linguistic questionnaire

Usage

GLOB.EVAL(
    Full_Database,
    MI,
    bmi,
    SI,
    b_jkt,
    p_ind = rep(1/nrow(Full_Database), nrow(Full_Database)),
    distance.type,
    i = 1,
    j = 1,
    theta = 1/3,
    thetas = 1,
    p = 2,
    q = 0.5,
    breakpoints = 100
)
Arguments

- **Full_Database**: the data set to evaluate.
- **MI**: a numerical value representing the total number of main-items dividing the linguistic questionnaire.
- **bmi**: an array referring to the initial weights of the main-items.
- **SI**: an array representing the total numbers of sub-items per main-item.
- **b_jkt**: a matrix of MI rows and max(SI) columns expressing the initial weights of each sub-item of a given main-item.
- **p_ind**: a vector of the relative sampling weights of the units, for which \( \sum_{i=1}^{n} p_{ind} \). If the weights are not relative, the following expression should be applied on the vector:

\[
\frac{p_{ind}}{\sum_{i=1}^{n} p_{ind}}
\]

If no sampling weights are used, the vector of weights is reduced to a vector of values 1, i.e. \( \text{rep}(1, nrow(data)) \).

- **distance.type**: type of distance chosen from the family of distances, set by default to the signed distance. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhopp", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
- **i**: parameter of the density function of the Beta distribution, fixed by default to \( i = 1 \).
- **j**: parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).
- **theta**: a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
- **thetas**: a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
- **p**: a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the Rho_p and Delta_pq.
- **q**: a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A data set of individual evaluations, for which the number of observations is exactly the same as the initial data set.

Examples

```r
data <- matrix(c(3,4,2,3,3,2,4,3,3,4,3,4,2,5,3,4,4,3,3,4,4,3,3,4,4,3,
   3,3,4,3,3,4,4,3,5,3,4,3,3,3), ncol = 4)
```
data <- as.data.frame(data)
MI <- 2
SI1 <- 2
SI2 <- 2
SI <- c(SI1, SI2)
b_j <- c(1/2, 1/2)
b_jk <- matrix(c(0.5, 0.5, 0.5, 0.5), nrow=2)
PA11 <- c(1, 2, 3, 4, 5)
PA12 <- c(1, 2, 3, 4, 5)
PA21 <- c(1, 2, 3, 4, 5)
PA22 <- c(1, 2, 3, 4, 5)
# ------------------
MF111 <- TrapezoidalFuzzyNumber(0, 2, 2, 7)
MF112 <- TrapezoidalFuzzyNumber(2, 7, 7, 15)
MF113 <- TrapezoidalFuzzyNumber(7, 15, 15, 23)
MF114 <- TrapezoidalFuzzyNumber(15, 23, 23, 28)
MF115 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF11 <- GFUZZ(data, 1, 1, PA11, spec="Identical", breakpoints = 100)
# ------------------
MF121 <- TrapezoidalFuzzyNumber(0, 2, 2, 7)
MF122 <- TrapezoidalFuzzyNumber(2, 7, 7, 15)
MF123 <- TrapezoidalFuzzyNumber(7, 15, 15, 23)
MF124 <- TrapezoidalFuzzyNumber(15, 23, 23, 28)
MF125 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF12 <- GFUZZ(data, 1, 2, PA12, spec="Identical", breakpoints = 100)
# ------------------
MF211 <- TrapezoidalFuzzyNumber(0, 2, 2, 7)
MF212 <- TrapezoidalFuzzyNumber(2, 7, 7, 15)
MF213 <- TrapezoidalFuzzyNumber(7, 15, 15, 23)
MF214 <- TrapezoidalFuzzyNumber(15, 23, 23, 28)
MF215 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF21 <- GFUZZ(data, 2, 1, PA21, spec="Identical", breakpoints = 100)
# ------------------
MF221 <- TrapezoidalFuzzyNumber(0, 2, 2, 7)
MF222 <- TrapezoidalFuzzyNumber(2, 7, 7, 15)
MF223 <- TrapezoidalFuzzyNumber(7, 15, 15, 23)
MF224 <- TrapezoidalFuzzyNumber(15, 23, 23, 28)
MF225 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF22 <- GFUZZ(data, 2, 2, PA22, spec="Identical", breakpoints = 100)
# ------------------
range <- matrix(c(0, 0, 0, 0, 28, 28, 28, 28), ncol=2)
ind.eval <- IND.EVAL(data, MI, b_j, SI, b_jk, range = range, distance.type ="DSGD.G")
GLOB <- GLOB.EVAL(data, MI, b_j, SI, b_jk, distance.type ="GSGD")

---

**GLOB.EVAL.mean**

*Calculates the weighted mean of the set of individual evaluations*

**Description**

Calculates the weighted mean of the set of individual evaluations
Usage

GLOB.EVAL.mean(ind.eval, weight = rep(1, length(ind.eval)))

Arguments

ind.eval       the set of individual evaluations.
weight         a vector of the relative sampling weights of the units, for which length(weight) = length(ind.eval), set by default to rep(1, length(ind.eval)).

Value

An integer.

Examples

data <- matrix(c(3,4,2,3,2,4,3,4,3,4,4,4,2,5,3,4,4,3,3,4,4,3, 3,3,4,3,3,4,4,4,3,5,4,3,3,3), ncol = 4)
data <- as.data.frame(data)
MI <- 2
SI1 <- 2
SI2 <- 2
SI <- c(SI1,SI2)
b_j <- c(1/2,1/2)
b_jk <- matrix(c(0.5,0.5,0.5,0.5),nrow=2)
PA11 <- c(1,2,3,4,5)
PA12 <- c(1,2,3,4,5)
PA21 <- c(1,2,3,4,5)
PA22 <- c(1,2,3,4,5)
# ------------------
MF111 <- TrapezoidalFuzzyNumber(0,2,2,7)
MF112 <- TrapezoidalFuzzyNumber(2,7,7,15)
MF113 <- TrapezoidalFuzzyNumber(7,15,15,23)
MF114 <- TrapezoidalFuzzyNumber(15,23,23,28)
MF115 <- TrapezoidalFuzzyNumber(23,28,28,30)
MF11 <- GFUZZ(data, 1, 1, PA11, spec="Identical", breakpoints = 100)
# ------------------
MF121 <- TrapezoidalFuzzyNumber(0,2,2,7)
MF122 <- TrapezoidalFuzzyNumber(2,7,7,15)
MF123 <- TrapezoidalFuzzyNumber(7,15,15,23)
MF124 <- TrapezoidalFuzzyNumber(15,23,23,28)
MF125 <- TrapezoidalFuzzyNumber(23,28,28,30)
MF12 <- GFUZZ(data, 1, 2, PA12, spec="Identical", breakpoints = 100)
# ------------------
MF211 <- TrapezoidalFuzzyNumber(0,2,2,7)
MF212 <- TrapezoidalFuzzyNumber(2,7,7,15)
MF213 <- TrapezoidalFuzzyNumber(7,15,15,23)
MF214 <- TrapezoidalFuzzyNumber(15,23,23,28)
MF215 <- TrapezoidalFuzzyNumber(23,28,28,30)
MF21 <- GFUZZ(data, 2, 1, PA21, spec="Identical", breakpoints = 100)
# ------------------
MF221 <- TrapezoidalFuzzyNumber(0,2,2,7)
MF222 <- TrapezoidalFuzzyNumber(2, 7, 15)
MF223 <- TrapezoidalFuzzyNumber(7, 15, 23)
MF224 <- TrapezoidalFuzzyNumber(15, 23, 28)
MF225 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF22 <- FUZZ(data, 2, 2, PA22, spec="Identical", breakpoints = 100)
# ------------------
range <- matrix(c(0, 0, 0, 0, 28, 28, 28, 28), ncol=2)
ind.eval <- IND.EVAL(data, MI, b_j, SI, b_jk, range = range, distance.type = "DSGD.G")
GLOB.mean <- GLOB.EVAL.mean(ind.eval)

---

GSGD

Calculates a distance between fuzzy numbers

Description

Calculates a distance between fuzzy numbers

Usage

GSGD(X, Y, i = 1, j = 1, thetas = 1, breakpoints = 100)

Arguments

X

a fuzzy number.

Y

a fuzzy number.

i

parameter of the density function of the Beta distribution, fixed by default to i = 1.

j

parameter of the density function of the Beta distribution, fixed by default to j = 1.

thetas

a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

breakpoints

a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.
IND.EVAL

Calculates the individual evaluations of a linguistic questionnaire

Description

Calculates the individual evaluations of a linguistic questionnaire

Usage

IND.EVAL(
    Full_Database,
    MI,
    bmi,
    SI,
    b_jkt,
    range,
    distance.type,
    i = 1,
    j = 1,
    theta = 1/3,
    thetas = 1,
    p = 2,
    q = 0.5,
    breakpoints = 100,
    spec = "Identical"
)

Arguments

Full_Database  the data set to evaluate.
MI  a numerical value representing the total number of main-items dividing the linguistic questionnaire.
bmi  an array referring to the initial weights of the main-items.
SI  an array representing the total numbers of sub-items per main-item.
b_jkt  a matrix of MI rows and max(SI) columns expressing the initial weights of each sub-item of a given main-item.
range  a vector of 2 elements giving the range of definition of the produced individual evaluations. The range is usually chosen in the interval between 0 and the maximum of the support set of all the membership functions modelling the data set.
distance.type  type of distance chosen from the family of distances, set by default to the signed distance. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i  parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).

theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{\text{Bertoluzza}} \), \( d_{\text{mid/spr}} \) and \( d_{\text{phi-wabl/ldev/rdev}} \).

thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the \( d_{\text{theta}} \) star and the \( d_{\text{GSGD}} \) distances.

p a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( Rho_{p} \) and \( Delta_{pq} \).

q a decimal value between 0 and 1, referring to the parameter of the metric \( Delta_{pq} \).

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

spec specification of the fuzzification matrix. The possible values are "Identical" and "Not Identical".

Value

A data set of individual evaluations, for which the number of observations is exactly the same as the initial data set.

Examples

data <- as.data.frame(data)
MI <- 2
SI1 <- 2
SI2 <- 2
SI <- c(SI1,SI2)
b_j <- c(1/2,1/2)
b_jk <- matrix(c(0.5,0.5,0.5,0.5),nrow=2)
PA11 <- c(1,2,3,4,5)
PA12 <- c(1,2,3,4,5)
PA21 <- c(1,2,3,4,5)
PA22 <- c(1,2,3,4,5)
# ------------------
MF111 <- TrapezoidalFuzzyNumber(0,2,2,7)
MF112 <- TrapezoidalFuzzyNumber(2,7,7,15)
MF113 <- TrapezoidalFuzzyNumber(7,15,15,23)
MF114 <- TrapezoidalFuzzyNumber(15,23,23,28)
MF115 <- TrapezoidalFuzzyNumber(23,28,28,30)
MF11 <- GFUZZ(data, 1, 1, PA11, spec="Identical", breakpoints = 100)
# ------------------
MF121 <- TrapezoidalFuzzyNumber(0,2,2,7)
MF122 <- TrapezoidalFuzzyNumber(2,7,7,15)
MF123 <- TrapezoidalFuzzyNumber(7,15,15,23)
MF124 <- TrapezoidalFuzzyNumber(15,23,23,28)
MF125 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF12 <- GFUZZ(data, 1, 2, PA12, spec="Identical", breakpoints = 100)
# ------------------
MF211 <- TrapezoidalFuzzyNumber(0, 2, 2, 7)
MF212 <- TrapezoidalFuzzyNumber(2, 7, 7, 15)
MF213 <- TrapezoidalFuzzyNumber(7, 15, 15, 23)
MF214 <- TrapezoidalFuzzyNumber(15, 23, 23, 28)
MF215 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF21 <- GFUZZ(data, 2, 1, PA21, spec="Identical", breakpoints = 100)
# ------------------
MF221 <- TrapezoidalFuzzyNumber(0, 2, 2, 7)
MF222 <- TrapezoidalFuzzyNumber(2, 7, 7, 15)
MF223 <- TrapezoidalFuzzyNumber(7, 15, 15, 23)
MF224 <- TrapezoidalFuzzyNumber(15, 23, 23, 28)
MF225 <- TrapezoidalFuzzyNumber(23, 28, 28, 30)
MF22 <- GFUZZ(data, 2, 2, PA22, spec="Identical", breakpoints = 100)
# ------------------
range <- matrix(c(0, 0, 0, 0, 28, 28, 28, 28), ncol=2)
ind.eval <- IND.EVAL(data, MI, b_j, SI, b_jk, range = range, distance.type ="DSGD.G")

int.0

*Numerical integration by the trivial method - method 1*

**Description**

Numerical integration by the trivial method - method 1

**Usage**

int.0(cut, a = 0, b = 1)

**Arguments**

cut a vector.
a fixed by default to 0.
b fixed by default to 1.

**Value**

An integer.
int.ct

**Description**
Numerical integration by the composite trapezoidal method - method 3

**Usage**
int.ct(cut, a = 0, b = 1)

**Arguments**
cut a vector.
a fixed by default to 0.
b fixed by default to 1.

**Value**
An integer.

int.simpson

**Description**
Numerical integration by the Simpson method - method 4

**Usage**
int.simpson(alpha, cut, a = 0, b = 1)

**Arguments**
alpha a vector of alpha values between 0 and 1.
cut a vector.
a fixed by default to 0.
b fixed by default to 1.

**Value**
An integer.
int.t

Numerical integration - method 2

Description

Numerical integration - method 2

Usage

int.t(alpha, cut, a = 0, b = 1)

Arguments

alpha a vector of alpha values between 0 and 1.
cut a vector.
a fixed by default to 0.
b fixed by default to 1.

Value

An integer.

integrate.num

Numerical integration by a particular method

Description

Numerical integration by a particular method

Usage

integrate.num(alpha, cut, method, a = 0, b = 1)

Arguments

alpha a vector of alpha values between 0 and 1.
cut a vector.
method the integration method could be one of the following four methods: "int.0", "int.t", "int.ct" and "int.simpson".
a fixed by default to 0.
b fixed by default to 1.

Value

An integer.
is.alphacuts

Verifies if a matrix is set of left and right alpha-cuts

Description
Verifies if a matrix is set of left and right alpha-cuts

Usage
is.alphacuts(data)

Arguments
data

a matrix of 2 equal length columns with no NA.

Value
A value TRUE if the concerned object can be a set of numerical left and right alpha-cuts, FALSE otherwise.

Examples
mat <- matrix(c(1,2,3,7,6,5), ncol = 2)
is.alphacuts(mat)

is.balanced

Verifies if a design is balanced

Description
Verifies if a design is balanced

Usage
is.balanced(ni)

Arguments

ni

a line array given by the contingency table related to the considered variable. Often written as a result of a call of the function table.

Value
Returns a logical decision TRUE or FALSE, to indicate if a given design is respectively balanced or not.
is.fuzzification

Verifies if a matrix is a fuzzification matrix

Usage

is.fuzzification(data)

Arguments

data        an array of 3 dimensions c(m,n,2), with m lines, n columns. No NA are allowed.

Value

A value TRUE if the concerned object is a numerical fuzzification matrix, FALSE otherwise.

Examples

mat <- array(c(1,1,2,2,3,3,5,5,6,6,7,7),dim=c(2,3,2))
is.fuzzification(mat)

is.trfuzzification

Verifies if a matrix is a fuzzification matrix of trapezoidal fuzzy numbers

Usage

is.trfuzzification(data)

Arguments

data        a matrix of 4 columns (p,q,r,s), where p \leq q \leq r \leq s. No NA are allowed.
Kurtosis

Value

A value TRUE if the concerned object is a trapezoidal or triangular fuzzification matrix, FALSE otherwise.

Examples

mat <- matrix(c(1,1,2,2,3,3,4,4),ncol=4)
is.trfuzzification(mat)

---

Kurtosis

Calculates the excess of kurtosis of a random fuzzy variable

Description

Calculates the excess of kurtosis of a random fuzzy variable

Usage

Kurtosis(
  data.fuzzified,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
dist.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
**thetas**
a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

**p**
a positive integer such that $1 \leq p < \infty$, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

**q**
a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, q is fixed to 0.5.

**breakpoints**
a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**
A numerical value.

**Examples**
```r
mat <- matrix(c(1,2,0.25,1.8,2,2.6,0.5,3,3,2.6,3.8,4,4,4.2,3.9,5), ncol =4)
Kurtosis(mat, dist.type = "GSGD")
```

---

**Mid.Spr**
*Calculates a distance by the d_Mid.Spr between fuzzy numbers*

**Description**
Calculates a distance by the d_Mid.Spr between fuzzy numbers

**Usage**
```r
Mid.Spr(X, Y, i = 1, j = 1, theta = 1/3, breakpoints = 100)
```

**Arguments**
- **X**
a fuzzy number.
- **Y**
a fuzzy number.
- **i**
parameter of the density function of the Beta distribution, fixed by default to $i = 1$.
- **j**
parameter of the density function of the Beta distribution, fixed by default to $j = 1$.
- **theta**
a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
- **breakpoints**
a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**
A numerical value.
Moment

Calculates a central sample moment of a random fuzzy variable

Description

Calculates a central sample moment of a random fuzzy variable

Usage

```r
Moment(
  data.fuzzified,
  k,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)
```

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

k the order of the moment.

dist.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

i parameter of the density function of the Beta distribution, fixed by default to i = 1.

j parameter of the density function of the Beta distribution, fixed by default to j = 1.

theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.

thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, q is fixed to 0.5.
breakpoints  a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.

Examples

```r
mat <- matrix(c(1,2,3,3,4,4,5), ncol =4)
Moment(mat, k=4, dist.type = "GSGD")
```

---

**nbreakpoints**  
_Calculates the number of breakpoints of a numerical matrix of alpha-cuts_

Description

Calculates the number of breakpoints of a numerical matrix of alpha-cuts

Usage

```r
nbreakpoints(data)
```

Arguments

```r
data  a matrix of numerical alpha-cuts or a 3-dimensional array. No NA are allowed.
```

Value

A numerical positive integer.

Examples

```r
X <- TrapezoidalFuzzyNumber(1,2,3,4)
algebra.X <- alphacut(X, seq(0,1,0.01))
nbreakpoints(algebra.X)
```
Calculates the number of answers by a specific sub-item

**Description**

Calculates the number of answers by a specific sub-item

**Usage**

\[ n_{jk..}(x, \text{varindex}, \text{PA}, \text{p\_ind} = \text{rep}(1, \text{nrow}(x))) \]

**Arguments**

- **x**
  - the data set to evaluate.
- **varindex**
  - index of a particular sub-item.
- **PA**
  - set of possible linguistic terms.
- **p\_ind**
  - a vector of the relative sampling weights of the units, for which \( \text{length}(\text{p\_ind}) = \text{nrow}(\text{data}) \). If the weights are not relative, the following expression should be applied on the vector:

\[
\frac{\text{p\_ind}}{\sum_{i=1}^{n} \text{p\_ind}}.
\]

If no sampling weights are used, the vector of weights is reduced to a vector of values 1, i.e. \( \text{rep}(1, \text{nrow}(\text{data})) \).

**Value**

A positive integer.

Calculates the number of answers by a specific linguistic of a sub-item

**Description**

Calculates the number of answers by a specific linguistic of a sub-item

**Usage**

\[ n_{jkd..}(x, \text{varindex}, q, \text{p\_ind} = \text{rep}(1, \text{nrow}(x))) \]
Arguments

- **x**: the data set to evaluate.
- **varindex**: index of a particular sub-item.
- **q**: index of a particular linguistic term.
- **p_ind**: a vector of the relative sampling weights of the units, for which \( \text{length}(p\text{nd}) = n\text{row}(\text{data}) \). If the weights are not relative, the following expression should be applied on the vector:

\[
\frac{p\text{nd}}{\sum_{i=1}^{N} p\text{nd}}.
\]

If no sampling weights are used, the vector of weights is reduced to a vector of values 1, i.e. \( \text{rep}(1, n\text{row}(\text{data})) \).

Value

A positive integer.

---

**p.value.fisher**  
Calculates the p-value of fuzzy observations taken from a Fisher distribution

---

Description

Calculates the p-value of fuzzy observations taken from a Fisher distribution

Usage

```R
p.value.fisher(  
  type,  
  H0,  
  H1,  
  t,  
  n,  
  r,  
  s.d,  
  sig,  
  dist.type,  
  i = 1,  
  j = 1,  
  theta = 1/3,  
  thetas = 1,  
  p = 2,  
  q = 0.5,  
  breakpoints = 100  
)
```
Arguments

type  a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.

H0  a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.

H1  a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.

t  a given numerical or fuzzy type parameter of the distribution.

n  first degree of freedom.

r  second degree of freedom.

s.d  a numerical value for the standard deviation of the distribution.

sig  a numerical value representing the significance level of the test.

dist.type  type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

i  parameter of the density function of the Beta distribution, fixed by default to i = 1.

j  parameter of the density function of the Beta distribution, fixed by default to j = 1.

theta  a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: $d_{Bertoluzza}$, $d_{mid/spr}$ and $d_{phi-wabl/ldev/rdev}$.

thetas  a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the $d_{theta}$ star and the $d_{GSGD}$ distances.

p  a positive integer such that $1 \leq p < \infty$, referring to the parameter of the $Rho_p$ and $Delta.pq$. By default, p is fixed to 2.

q  a decimal value between 0 and 1, referring to the parameter of the metric $Delta.pq$. By default, p is fixed to 0.5.

breakpoints  a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns the defuzzified p-value and the decision made.
p.value.log

Calculates the p-value of fuzzy observations taken from a Logistic distribution

Description

Calculates the p-value of fuzzy observations taken from a Logistic distribution

Usage

p.value.log(
  type,  # a category between "0", "1", and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
  H0,  # a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
  H1,  # a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
  t,  # a given numerical or fuzzy type parameter of the distribution.
  n,  # the total number of observations of the data set.
  s.d,  # a numerical value for the standard deviation of the distribution.
  sig,  # a numerical value representing the significance level of the test.
  dist.type,  # type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
  i = 1,  # parameter of the density function of the Beta distribution, fixed by default to i = 1.
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)

Arguments

type
  a category between "0", "1", and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.

H0
  a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.

H1
  a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.

t
  a given numerical or fuzzy type parameter of the distribution.

n
  the total number of observations of the data set.

s.d
  a numerical value for the standard deviation of the distribution.

sig
  a numerical value representing the significance level of the test.

dist.type
  type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

i
  parameter of the density function of the Beta distribution, fixed by default to i = 1.
parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).

theta  

a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to \( \frac{1}{3} \) referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{\text{Bertoluzza}}, d_{\text{mid/spr}} \) and \( d_{\text{phi-wabl/ldev/rdev}} \).

thetas  
a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the \( d_{\text{theta star}} \) and the \( d_{\text{GSGD}} \) distances.

p  
a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( \text{Rho}_{p} \) and \( \text{Delta}_{pq} \). By default, p is fixed to 2.

q  
a decimal value between 0 and 1, referring to the parameter of the metric \( \text{Delta}_{pq} \). By default, p is fixed to 0.5.

breakpoints  
a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns the defuzzified p-value and the decision made.

---

### p.value.mean.log

**Calculates the p-value of fuzzy observations taken from a logistic distribution for the mean**

**Description**

Calculates the p-value of fuzzy observations taken from a logistic distribution for the mean

**Usage**

```r
p.value.mean.log(
  data.fuzzified,
  type,
  H0,
  H1,
  s.d,
  sig,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)
```
Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
type a category betweenn "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
H₀ a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
H₁ a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
s.d a numerical value for the standard deviation of the distribution.
sig a numerical value representing the significance level of the test.
dist.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
thetaa a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns the defuzzified p-value and the decision made.

---

**p.value.mean.normal**

*Calculates the p-value of fuzzy observations taken from a normal distribution for the mean*

---

**Description**

Calculates the p-value of fuzzy observations taken from a normal distribution for the mean
p.value.mean.normal

Usage

p.value.mean.normal(
  data.fuzzified,
  type,
  H0,
  H1,
  s.d,
  sig,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
type a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
H0 a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
H1 a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
s.d a numerical value for the standard deviation of the distribution.
sig a numerical value representing the significance level of the test.
dist.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
theta2 a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
\[ p \] a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the Rho\(_p\) and Delta\(_{pq}\). By default, \( p \) is fixed to 2.

\[ q \] a decimal value between 0 and 1, referring to the parameter of the metric Delta\(_{pq}\). By default, \( p \) is fixed to 0.5.

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

Returns the defuzzified p-value and the decision made.

---

**p.value.mean.poisson**

*Calculates the p-value of fuzzy observations taken from a Poisson distribution for the mean*

---

**Description**

Calculates the p-value of fuzzy observations taken from a Poisson distribution for the mean

**Usage**

\[
p\text{.value.mean.poisson(}
\text{data.fuzzified,}
\text{type,}
\text{H0,}
\text{H1,}
\text{sig,}
\text{dist.type,}
\text{i = 1,}
\text{j = 1,}
\text{theta = 1/3,}
\text{thetas = 1,}
\text{p = 2,}
\text{q = 0.5,}
\text{breakpoints = 100}
\text{)}
\]

**Arguments**

- **data.fuzzified** a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
- **type** a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
- **H0** a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
**p.value.mean.Student**

Calculates the p-value of fuzzy observations taken from a Student distribution for the mean

**Description**

Calculates the p-value of fuzzy observations taken from a Student distribution for the mean

**Usage**

```r
p.value.mean.Student(
  data.fuzzified,
  type,
  H0,
  H1,
  sig,
)```

- `data.fuzzified`: A data frame containing the fuzzy observations.
- `type`: A character string indicating the type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
- `H0`: A trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
- `H1`: A numerical value representing the significance level of the test.
- `sig`: A numerical value representing the significance level of the test.
- `dist.type`: A numerical value representing the significance level of the test.
- `i`: Parameter of the density function of the Beta distribution, fixed by default to i = 1.
- `j`: Parameter of the density function of the Beta distribution, fixed by default to j = 1.
- `theta`: A numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
- `thetas`: A decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
- `p`: A positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
- `q`: A decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
- `breakpoints`: A positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

Returns the defuzzified p-value and the decision made.
p.value.mean.Student

dist.type,
i = 1,
j = 1,
theta = 1/3,
thetas = 1,
p = 2,
q = 0.5,
breakpoints = 100
)

Arguments

data.fuzzified a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
type a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
H0 a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
H1 a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
sig a numerical value representing the significance level of the test.
dist.type type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i parameter of the density function of the Beta distribution, fixed by default to i = 1.
j parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
thetas a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
p a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

Returns the defuzzified p-value and the decision made.
p.value.normal

Calculates the p-value of fuzzy observations taken from a normal distribution

Description

Calculates the p-value of fuzzy observations taken from a normal distribution

Usage

p.value.normal(
  type,
  H0,
  H1,
  t,
  n,
  s.d,
  sig,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>a category between &quot;0&quot;, &quot;1&quot; and &quot;2&quot;. The category &quot;0&quot; refers to a bilateral test, the category &quot;1&quot; for a lower unilateral one, and &quot;2&quot; for an upper unilateral test.</td>
</tr>
<tr>
<td>H0</td>
<td>a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.</td>
</tr>
<tr>
<td>H1</td>
<td>a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.</td>
</tr>
<tr>
<td>t</td>
<td>a given numerical or fuzzy type parameter of the distribution.</td>
</tr>
<tr>
<td>n</td>
<td>the total number of observations of the data set.</td>
</tr>
<tr>
<td>s.d</td>
<td>a numerical value for the standard deviation of the distribution.</td>
</tr>
<tr>
<td>sig</td>
<td>a numerical value representing the significance level of the test.</td>
</tr>
<tr>
<td>dist.type</td>
<td>type of distance chosen from the family of distances. The different choices are given by: &quot;Rho1&quot;, &quot;Rho2&quot;, &quot;Bertoluzza&quot;, &quot;Rhop&quot;, &quot;Delta.pq&quot;, &quot;Mid/Spr&quot;, &quot;wabl&quot;, &quot;DSGD&quot;, &quot;DSGD.G&quot;, &quot;GSGD&quot;.</td>
</tr>
<tr>
<td>i</td>
<td>parameter of the density function of the Beta distribution, fixed by default to i = 1.</td>
</tr>
</tbody>
</table>
**p.value.poisson**

- **j**
  - Parameter of the density function of the Beta distribution, fixed by default to $j = 1$.

- **theta**
  - A numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to $1/3$ referring to the Lebesgue space. This measure is used in the calculations of the following distances: $d_{	ext{Bertoluzza}}$, $d_{	ext{mid/spr}}$ and $d_{	ext{phi-wabl/ldev/rdev}}$.

- **thetas**
  - A decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the $d_{\text{theta star}}$ and the $d_{\text{GSGD}}$ distances.

- **p**
  - A positive integer such that $1 \leq p < \infty$, referring to the parameter of the $\text{Rho}_p$ and $\text{Delta}_pq$. By default, p is fixed to 2.

- **q**
  - A decimal value between 0 and 1, referring to the parameter of the metric $\text{Delta}_pq$. By default, p is fixed to 0.5.

- **breakpoints**
  - A positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

Returns the defuzzified p-value and the decision made.

---

**p.value.poisson**

Calculates the p-value of fuzzy observations taken from a Poisson distribution.

**Description**

Calculates the p-value of fuzzy observations taken from a Poisson distribution.

**Usage**

```r
p.value.poisson(  type,  H0,  H1,  t,  n,  sig,  dist.type,  i = 1,  j = 1,  theta = 1/3,  thetas = 1,  p = 2,  q = 0.5,  breakpoints = 100,  s.d = 1  )
```
Arguments

- **type**: a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.

- **H0**: a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.

- **H1**: a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.

- **t**: a given numerical or fuzzy type parameter of the distribution.

- **n**: the total number of observations of the data set.

- **sig**: a numerical value representing the significance level of the test.

- **dist.type**: type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

- **i**: parameter of the density function of the Beta distribution, fixed by default to i = 1.

- **j**: parameter of the density function of the Beta distribution, fixed by default to j = 1.

- **theta**: a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.

- **thetas**: a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

- **p**: a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

- **q**: a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, p is fixed to 0.5.

- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

- **s.d**: a numerical value for the standard deviation of the distribution.

Value

Returns the defuzzified p-value and the decision made.
Calculates the p-value of fuzzy observations taken from a Student distribution

Description

Calculates the p-value of fuzzy observations taken from a Student distribution

Usage

```r
p.value.Student(
  type,
  H0,
  H1,
  t,
  n,
  sig,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100,
  s.d = 1
)
```

Arguments

- `type`: a category between "0", "1" and "2". The category "0" refers to a bilateral test, the category "1" for a lower unilateral one, and "2" for an upper unilateral test.
- `H0`: a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
- `H1`: a trapezoidal or a triangular fuzzy number representing the fuzzy alternative hypothesis.
- `t`: a given numerical or fuzzy type parameter of the distribution.
- `n`: the total number of observations of the data set.
- `sig`: a numerical value representing the significance level of the test.
- `dist.type`: type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
- `i`: parameter of the density function of the Beta distribution, fixed by default to i = 1.
\( j \) is a parameter of the density function of the Beta distribution, fixed by default to \( j = 1 \).

\( \theta \) is a numerical value between 0 and 1, representing a weighting parameter. By default, \( \theta \) is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \( d_{\text{Bertoluzza}} \), \( d_{\text{mid/spr}} \) and \( d_{\phi\text{-wabl/ldev/rdev}} \).

\( \theta_s \) is a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, \( \theta_s \) is fixed to 1. This parameter is used in the calculations of the \( d_{\theta \text{-star}} \) and the \( d_{\text{GS GD}} \) distances.

\( p \) is a positive integer such that \( 1 \leq p < \infty \), referring to the parameter of the \( Rho_p \) and \( Delta_{pq} \). By default, \( p \) is fixed to 2.

\( q \) is a decimal value between 0 and 1, referring to the parameter of the metric \( Delta_{pq} \). By default, \( q \) is fixed to 0.5.

\( \text{breakpoints} \) is a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

\( s.d \) is a numerical value for the standard deviation of the distribution.

**Value**

Returns the defuzzified \( p \)-value and the decision made.

---

\[ R \]

*Calculates the indicator of information's rate of the data base*

**Description**

Calculates the indicator of information’s rate of the data base

**Usage**

\[
R(x, \ p\_ind, \ b\_jk, \ SI)
\]

**Arguments**

- \( x \) is the data set to evaluate.
- \( p\_ind \) is a vector of the relative sampling weights of the units, for which \( \text{length}(p\_ind) = \text{nrow}(data) \). If the weights are not relative, the following expression should be applied on the vector:
  \[
  \frac{P_{ind}}{\sum_{i=1}^{n} P_{ind}}
  \]
  If no sampling weights are used, the vector of weights is reduced to a vector of values 1, i.e. \( \text{rep}(1, \text{nrow}(data)) \).
- \( b\_jk \) is a matrix of length(\( b\_j \)) rows and \( \text{max}(SI) \) columns expressing the initial weights of each sub-item of a given main-item.
- \( SI \) is an array representing the total numbers of sub-items per main-item.
Value

A numerical value giving the indicator of information’s rate of the complete linguistic questionnaire. Note that the obtained value is interpreted as the more it tends to the value 1, the less the complete questionnaire contains missing values.

Examples

data <- as.data.frame(data)
p_ind <- c(0.1,0.05,0.05,0.2,0.1,0.05,0.1,0.1,0.2,0.05)
SI1 <- 2
SI2 <- 2
SI <- c(SI1,SI2)
b_jk <- matrix(c(0.5,0.5,0.5,0.5),nrow=2)
R(data, p_ind, b_jk, SI)

Rho1

Calculates a distance by the Rho1 between fuzzy numbers

Description

Calculates a distance by the Rho1 between fuzzy numbers

Usage

Rho1(X, Y, breakpoints = 100)

Arguments

X a fuzzy number.
Y a fuzzy number.
breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.
Rho2

*Calculates a distance by the Rho2 between fuzzy numbers*

Description

Calculates a distance by the Rho2 between fuzzy numbers

Usage

Rho2(X, Y, breakpoints = 100)

Arguments

- **X**: a fuzzy number.
- **Y**: a fuzzy number.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.

Rhop

*Calculates a distance by the d_Rhop between fuzzy numbers*

Description

Calculates a distance by the d_Rhop between fuzzy numbers

Usage

Rhop(X, Y, p, breakpoints = 100)

Arguments

- **X**: a fuzzy number.
- **Y**: a fuzzy number.
- **p**: a positive integer such that $1 \leq p < \infty$, referring to the parameter of the $\text{Rho}_p$ and $\text{Delta}_{pq}$.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.
Ri

Calculates the indicator of information’s rate of the data base for a given unit

Description

Calculates the indicator of information’s rate of the data base for a given unit

Usage

Ri(x, i, b_jk, SI)

Arguments

x
  the data set to evaluate.

i
  an observation index.

b_jk
  a matrix of length(b_j) rows and max(SI) columns expressing the initial weights of each sub-item of a given main-item.

SI
  an array representing the total numbers of sub-items per main-item.

Value

A numerical value giving the indicator of information’s rate of the complete linguistic questionnaire for a particular observation. Note that the obtained value is interpreted as the more it tends to the value 1, the less the observation i contains missing values.

Examples

data <- as.data.frame(data)
SI1 <- 2
SI2 <- 2
SI <- c(SI1, SI2)
b_jk <- matrix(c(0.5, 0.5, 0.5, 0.5), nrow=2)
Ri(data, 7, b_jk, SI)
Sample.variance

Calculates the sample variance by a convenient metric

Description

Calculates the sample variance by a convenient metric

Usage

Sample.variance(
  data.fuzzified,
  dist.type,
  i = 1,
  j = 1,
  theta = 1/3,
  thetas = 1,
  p = 2,
  q = 0.5,
  breakpoints = 100
)

Arguments

data.fuzzified   a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.
dist.type        type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".
i               parameter of the density function of the Beta distribution, fixed by default to i = 1.
j               parameter of the density function of the Beta distribution, fixed by default to j = 1.
theta           a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: \(d_{\text{Bertoluzza}}\), \(d_{\text{mid/spr}}\) and \(d_{\text{phi-wabl/ldev/rdev}}\).
theta.s         a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the \(d_{\text{theta star}}\) and the \(d_{\text{GSGD}}\) distances.
p               a positive integer such that 1 \(\leq p < \) infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.
q               a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, q is fixed to 0.5.
breakpoints     a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
SEQ.ORDERING

Calculates the sequential sums of squares by a convenient metric

Description
Calculates the sequential sums of squares by a convenient metric

Usage
SEQ.ORDERING(scope, data, f.response)

Arguments
- scope: a description of the complete fitting model.
- data: the data frame containing all the variables of the model.
- f.response: the vector of distances of the fuzzy response variable to the fuzzy origin.

Value
Returns a list of the new sets of sums of squares, as well as the coefficients, the residuals and the fitted.values.

SEQ.ORDERING.APPROXIMATION

Calculates the sequential sums of squares by an approximation

Description
Calculates the sequential sums of squares by an approximation

Usage
SEQ.ORDERING.APPROXIMATION(scope, data, f.response)

Arguments
- scope: a description of the complete fitting model.
- data: the data frame containing all the variables of the model.
- f.response: the vector of distances of the fuzzy response variable to the fuzzy origin.

Value
Returns a list of the new sets of sums of squares, as well as the coefficients, the residuals and the fitted.values.
**SEQ.ORDERING.EXACT**

*Calculates the sequential sums of squares by an exact calculation*

**Description**

Calculates the sequential sums of squares by an exact calculation

**Usage**

SEQ.ORDERING.EXACT(scope, data, f.response)

**Arguments**

- **scope**: a description of the complete fitting model.
- **data**: the data frame containing all the variables of the model.
- **f.response**: the vector of distances of the fuzzy response variable to the fuzzy origin.

**Value**

Returns a list of the new sets of sums of squares, as well as the coefficients, the residuals and the fitted.values.

---

**SGD**

*Calculates a distance by the SGD between fuzzy numbers*

**Description**

Calculates a distance by the SGD between fuzzy numbers

**Usage**

SGD(X, i = 1, j = 1, breakpoints = 100)

**Arguments**

- **X**: a fuzzy number.
- **i**: parameter of the density function of the Beta distribution, fixed by default to i = 1.
- **j**: parameter of the density function of the Beta distribution, fixed by default to j = 1.
- **breakpoints**: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

**Value**

A numerical value.
**Skewness**  
*Calculates the skewness of a random fuzzy variable*

**Description**

Calculates the skewness of a random fuzzy variable

**Usage**

```r
Skewness(
    data.fuzzified,
    dist.type,
    i = 1,
    j = 1,
    theta = 1/3,
    thetas = 1,
    p = 2,
    q = 0.5,
    breakpoints = 100
)
```

**Arguments**

- **data.fuzzified**  
a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

- **dist.type**  
type of distance chosen from the family of distances. The different choices are given by: "Rho1", "Rho2", "Bertoluzza", "Rhop", "Delta.pq", "Mid/Spr", "wabl", "DSGD", "DSGD.G", "GSGD".

- **i**  
parameter of the density function of the Beta distribution, fixed by default to i = 1.

- **j**  
parameter of the density function of the Beta distribution, fixed by default to j = 1.

- **theta**  
a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.

- **thetas**  
a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, thetas is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.

- **p**  
a positive integer such that 1 ≤ p < infinity, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 2.

- **q**  
a decimal value between 0 and 1, referring to the parameter of the metric Delta_pq. By default, q is fixed to 0.5.

- **breakpoints**  
a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
Value

A numerical value.

Examples

mat <- matrix(c(1,2,0.25,1.8,2,2.6,0.5,3,3,2.6,3.8,4,4,4.2,3.9,5), ncol =4)
Skewness(mat, dist.type = "GSGD")

---

square

Square a number

---

Description

Takes any numerical value and squares it.

Usage

square(x)

Arguments

x

A numeric value to be squared

Value

The square of the input

---

tr.gfuzz

Fuzzifies a variable modelled by trapezoidal or triangular fuzzy numbers

---

Description

Fuzzifies a variable modelled by trapezoidal or triangular fuzzy numbers

Usage

tr.gfuzz(data, breakpoints = 100)

Arguments

data

a matrix of 4 columns (p,q,r,s), where p ≤ q ≤ r ≤ s. No NA are allowed.

breakpoints

a positive arbitrary integer representing the number of breaks chosen to build
the numerical alpha-cuts. breakpoints is fixed to 100 by default.
Value

A 3-dimensional array with dimensions (m,n,2), i.e. m lines, n columns, with no NA.

Examples

data <- matrix(c(1,1,2,2,3,3,4,4),ncol=4)
data.tr <- tr.gfuzz(data)

wabl

Calculates a distance by the d_wabl between fuzzy numbers

Description

Calculates a distance by the d_wabl between fuzzy numbers

Usage

wabl(X, Y, i = 1, j = 1, theta = 1/3, breakpoints = 100)

Arguments

X a fuzzy number.

Y a fuzzy number.

i parameter of the density function of the Beta distribution, fixed by default to i = 1.

j parameter of the density function of the Beta distribution, fixed by default to j = 1.

theta a numerical value between 0 and 1, representing a weighting parameter. By default, theta is fixed to 1/3 referring to the Lebesgue space. This measure is used in the calculations of the following distances: d_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.

breakpoints a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

Value

A numerical value.
Description

Calculates the weighted fuzzy sample mean

Usage

\[
\text{Weighted.fuzzy.mean(}
\begin{array}{l}
data.fuzzified, \\
\text{weight,} \\
\text{breakpoints} = 100, \\
\text{alphacuts} = \text{FALSE}\end{array}
\)
\]

Arguments

data.fuzzified: a fuzzification matrix constructed by a call to the function FUZZ or the function GFUZZ, or a similar matrix. No NA are allowed.

weight: a weighting vector of the same length of the fuzzification matrix. No NA allowed.

breakpoints: a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.

alphacuts: fixed by default to "FALSE". No alpha-cuts are printed in this case.

Value

If the parameter alphacuts="TRUE", the function returns a matrix composed by 2 vectors representing the numerical left and right alpha-cuts. For this output, is.alphacuts = TRUE. If the parameter alphacuts="FALSE", the function returns a trapezoidal fuzzy number given by the quadruple (p,q,r,s).

Examples

\[
\text{mat} \leftarrow \text{matrix}(c(1,2,2,3,4,4,5), \text{ncol=4)} \\
\text{w} \leftarrow \text{c(1,3)} \\
\text{Weighted.fuzzy.mean(mat, w)}
\]
## Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjusted.weight.MI</td>
<td>4</td>
</tr>
<tr>
<td>adjusted.weight.SI</td>
<td>5</td>
</tr>
<tr>
<td>Bertoluzza</td>
<td>6</td>
</tr>
<tr>
<td>boot.mean.algo1</td>
<td>7</td>
</tr>
<tr>
<td>boot.mean.algo2</td>
<td>8</td>
</tr>
<tr>
<td>boot.mean.ml</td>
<td>9</td>
</tr>
<tr>
<td>cube</td>
<td>10</td>
</tr>
<tr>
<td>D2</td>
<td>11</td>
</tr>
<tr>
<td>Defuzz.FANOVA</td>
<td>11</td>
</tr>
<tr>
<td>Delta.pq</td>
<td>12</td>
</tr>
<tr>
<td>Delta_jki</td>
<td>13</td>
</tr>
<tr>
<td>distance</td>
<td>14</td>
</tr>
<tr>
<td>DSGD</td>
<td>15</td>
</tr>
<tr>
<td>DSGD.G</td>
<td>16</td>
</tr>
<tr>
<td>FANOVA</td>
<td>16</td>
</tr>
<tr>
<td>FANOVA.approximation</td>
<td>18</td>
</tr>
<tr>
<td>FANOVA.distance</td>
<td>19</td>
</tr>
<tr>
<td>FANOVA.exact</td>
<td>20</td>
</tr>
<tr>
<td>FANOVA.summary</td>
<td>21</td>
</tr>
<tr>
<td>fci.ml</td>
<td>22</td>
</tr>
<tr>
<td>fci.ml.boot</td>
<td>23</td>
</tr>
<tr>
<td>FMANOVA</td>
<td>24</td>
</tr>
<tr>
<td>FMANOVA.approximation</td>
<td>26</td>
</tr>
<tr>
<td>FMANOVA.distance</td>
<td>27</td>
</tr>
<tr>
<td>FMANOVA.exact</td>
<td>28</td>
</tr>
<tr>
<td>FMANOVA.interaction.summary</td>
<td>29</td>
</tr>
<tr>
<td>FMANOVA.summary</td>
<td>30</td>
</tr>
<tr>
<td>Ftests</td>
<td>30</td>
</tr>
<tr>
<td>FTukeyHSD</td>
<td>31</td>
</tr>
<tr>
<td>FUZZ</td>
<td>32</td>
</tr>
<tr>
<td>Fuzzy.CI.ML.test</td>
<td>33</td>
</tr>
<tr>
<td>Fuzzy.CI.test</td>
<td>35</td>
</tr>
<tr>
<td>Fuzzy.decisions</td>
<td>37</td>
</tr>
<tr>
<td>Fuzzy.decisions.ML</td>
<td>38</td>
</tr>
<tr>
<td>Fuzzy.Difference</td>
<td>40</td>
</tr>
<tr>
<td>Fuzzy.exact.variance</td>
<td>41</td>
</tr>
<tr>
<td>Fuzzy.exact.variance.poly.left</td>
<td>42</td>
</tr>
<tr>
<td>Fuzzy.exact.variance.poly.right</td>
<td>42</td>
</tr>
<tr>
<td>Fuzzy.p.value</td>
<td>43</td>
</tr>
<tr>
<td>Fuzzy.p.value.mean</td>
<td>44</td>
</tr>
<tr>
<td>fuzzy.predicted.values</td>
<td>46</td>
</tr>
<tr>
<td>fuzzy.residuals</td>
<td>47</td>
</tr>
<tr>
<td>Fuzzy.sample.mean</td>
<td>47</td>
</tr>
<tr>
<td>Fuzzy.sample.variance.approximation</td>
<td>48</td>
</tr>
<tr>
<td>Fuzzy.sample.variance.approximation1</td>
<td>48</td>
</tr>
<tr>
<td>Fuzzy.sample.variance.approximation2</td>
<td>49</td>
</tr>
<tr>
<td>Fuzzy.sample.variance.approximation3</td>
<td>49</td>
</tr>
<tr>
<td>Fuzzy.sample.variance.approximation4</td>
<td>50</td>
</tr>
<tr>
<td>Fuzzy.sample.variance.approximation5</td>
<td>50</td>
</tr>
<tr>
<td>Fuzzy.Square</td>
<td>51</td>
</tr>
<tr>
<td>Fuzzy.Square.poly.left</td>
<td>51</td>
</tr>
<tr>
<td>Fuzzy.Square.poly.right</td>
<td>52</td>
</tr>
<tr>
<td>Fuzzy.variance</td>
<td>53</td>
</tr>
<tr>
<td>GaussianBellFuzzyNumber</td>
<td>54</td>
</tr>
<tr>
<td>GaussianFuzzyNumber</td>
<td>55</td>
</tr>
<tr>
<td>GFUZZ</td>
<td>56</td>
</tr>
<tr>
<td>GLOB.EVAL</td>
<td>57</td>
</tr>
<tr>
<td>GLOB.EVAL.mean</td>
<td>59</td>
</tr>
<tr>
<td>GSGD</td>
<td>61</td>
</tr>
<tr>
<td>IND.EVAL</td>
<td>62</td>
</tr>
<tr>
<td>int.0</td>
<td>64</td>
</tr>
<tr>
<td>int.ct</td>
<td>65</td>
</tr>
<tr>
<td>int.simpson</td>
<td>65</td>
</tr>
<tr>
<td>int.t</td>
<td>66</td>
</tr>
<tr>
<td>integrate.num</td>
<td>66</td>
</tr>
<tr>
<td>is.alphacuts</td>
<td>67</td>
</tr>
<tr>
<td>is.balanced</td>
<td>67</td>
</tr>
<tr>
<td>is.fuzzification</td>
<td>68</td>
</tr>
</tbody>
</table>
is.trfuzzification, 68
Kurtosis, 69
Mid.Spr, 70
Moment, 71
n_jk..., 73
n_jkq.., 73
nbbreakpoints, 72
p.value.fisher, 74
p.value.log, 76
p.value.mean.log, 77
p.value.mean.normal, 78
p.value.mean.poisson, 80
p.value.mean.Student, 81
p.value.normal, 83
p.value.poisson, 84
p.value.Student, 86
R, 87
Rho1, 88
Rho2, 89
Rhop, 89
Ri, 90
Sample.variance, 91
SEQ.ORDERING, 92
SEQ.ORDERING.APPROXIMATION, 92
SEQ.ORDERING.EXACT, 93
SGD, 93
Skewness, 94
square, 95
tr.gfuzz, 95
wabl, 96
Weighted.fuzzy.mean, 97