Package ‘FuzzySTs’

October 12, 2022

**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
- fci.ml ............................................................... 22
- fci.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
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
**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**

```r
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(3,4,2,3,2,4,3,3,4,4,2,5,3,4,4,3,3,4,4,4,3,3,4,4,3,3,2,3), 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.
boots.mean.algo1

---

**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,  # 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 = 100,      # an integer giving the number of replications needed in the bootstrap procedure. It is set to 100 by default.
  mu = NA,         # if the mean of the normal distribution is known, mu should be a numerical value. Otherwise, the argument mu is fixed to NA.
  sigma = 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.
  step = 0.1,      # a numerical value fixed to 0.1, defining the step of iterations on the interval [t-5; t+5].
  margin = c(5, 5),  # an optional numerical couple of values fixed to [5; 5], representing the range of calculations around the parameter t.
  breakpoints = 100,  # a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
  plot = TRUE       # fixed by default to "FALSE". plot="FALSE" if a plot of the fuzzy number is not required.
)
```

**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.

| 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
```r
boot.mean.algo2(
  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.

---

**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**

```r
boot.mean.ml(
  data.fuzzified,
  algorithm,
  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.
- `algorithm`: an algorithm chosen between "algo1" or "algo2".
- `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.

**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
Calculates a distance by the D2 between fuzzy numbers

**Usage**

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

**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.

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
)
```
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: $d_{\text{Bertoluzza}}$, $d_{\text{mid/spr}}$ 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 $\text{Delta}_p$.q. By default, $p$ is fixed to 2.

**q**  
A decimal value between 0 and 1, referring to the parameter of the metric $\text{Delta}_p$.q. 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 $d_{\text{Delta.pq}}$ between fuzzy numbers*

Description

Calculates a distance by the $d_{\text{Delta.pq}}$ 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 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 numerical value.

---

**Delta_jki**  
*Calculates the factor Delta_jki*

Description

Calculates the factor Delta_jki

Usage

`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_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.
- **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

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

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.
**DSGD.G**

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", "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.

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

```r
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(0,1,1,2.2)
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")
```
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: $d_{\text{Bertoluzza}}$, $d_{\text{mid/spr}}$ 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. 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
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

```r
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(1,2,3,2,1,3,1,2),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(1,2,3,2,1,1,3,1,2),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.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 FUZZZ 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 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(2,2,1,2,2,2,2,2,2,1,1,1,2,3,4,3,1,2,5,4,3),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")
detach(data)
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

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", "Rhopt", "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: \( d_{\text{Bertoluzza}} \), \( d_{\text{mid/spr}} \) 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, \( \theta_{\text{tas}} \) is fixed to 1. This parameter is used in the calculations of the \( d_{\theta \ast} \) 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 \( Delta_{pq} \). 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**

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.
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

Ftests

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

Description

Calculates multiple tests corresponding to the fuzzy response variable

Usage

Ftests(test)

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

Ftests

Calculates multiple tests corresponding to the fuzzy response variable
FTukeyHSD

Value

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

Examples

```
mat <- matrix(c(2,2,1,2,2,2,2,1,1,1,2,3,4,3,1,2,5,4,3),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(2,2,1,2,2,2,2,2,2,2,2,2,2,2,2,1,1,1,2,3,4,4,3,1,2,5,4,4,3),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**

```
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(1,2,3,2,1,3,1,2),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

Fuzzy.CI.ML.test(
  data.fuzzified,
  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 = NA,  # if the mean of the normal distribution is known, mu should be a numerical value.
  sigma = NA,# otherwise, the argument mu is fixed to 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.
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".

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

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.

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.

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

Value

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

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)
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,
  H0,
  H1,
  t,
  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.
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.
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

H0 <- TriangularFuzzyNumber(2.9,3,3.1)
H1 <- TriangularFuzzyNumber(3,3,5)
res <- Fuzzy.CI.test(type = 0, H0, H1, t = TriangularFuzzyNumber(0.8,1.8,2.80), 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**

```r
Fuzzy.decisions(
  type,
  H0,
  H1,
  t,
  s.d,
  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".
Fuzzy.decisions.ML

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

Description

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

Usage

Fuzzy.decisions.ML(
  data.fuzzified,
  H0,
  H1,
  type = 0,
  s.d = 0.79,
  n = 10,
  sig = 0.05,
  distribution = "normal",
  distance.type = "GSGD"
)
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/1dev/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.

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.

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

Value

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

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)
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))
coef.boot <- 3.470085
res <- Fuzzy.decisions.ML(data.fuzzified, H0, H1, t = t, coef.boot = coef.boot,
sigma = 0.79, sig = 0.05, distribution = "normal", distance.type = "GSGD")

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 \(p \leq q \leq r \leq s\).

Examples

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

Description

Calculates the exact variance

Usage

```r
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

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.
Fuzzy.p.value.mean

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.
theta  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.

Examples

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

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, 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.

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)
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
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
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.
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

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

Description

Fuzzy sample variance (approx) - general

Usage

```r
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.

Description

Fuzzy sample variance (approx) - method 1

Usage

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

Fuzzy sample variance (approx) - method 2

Description
Fuzzy sample variance (approx) - method 2

Usage
Fuzzy.sample.variance.approximation2(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.approximation3

Fuzzy sample variance (approx) - method 3

Description
Fuzzy sample variance (approx) - method 3

Usage
Fuzzy.sample.variance.approximation3(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.
**Fuzzy.sample.variance.approximation4**

*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.approximation5**

*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**

```r
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**

```r
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**

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

---

**Fuzzy.Square.poly.right**

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

**Description**

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

**Usage**

```r
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**

```r
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
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 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
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

```r
GaussianFuzzyNumber(
  mean,
  sigma,
  alphacuts = FALSE,
  margin = c(5, 5),
)```
step = 0.01,
breakpoints = 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 repre-
senting the left and right alpha-cuts. For this output, is.alphacuts = TRUE. If the parameter alpha-
cuts="FALSE", the function returns a list composed by the Class, the mean, the sigma, the vectors
of the left and right alpha-cuts.

Examples

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

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

Calculates the global evaluation of a linguistic questionnaire

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 \( \text{length}(p_{\text{ind}}) = nrow(data) \). If the weights are not relative, the following expression should be applied on the vector:

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

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", "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.

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 <- 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 \texttt{length(weight)} = \texttt{length(ind.eval)}, set by default to \texttt{rep(1,length(ind.eval))}.

Value

An integer.

Examples

data <- matrix(c(3,4,2,3,2,4,3,3,4,3,4,4,2,5,3,4,4,3,3,4,4,3,3,3,3, 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(7,15,15,23)
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.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**

```r
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 \( j = 1 \).

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

\( \text{thetas} \) a decimal value between 0 and 1, representing the weight given to the shape of the fuzzy number. By default, \( \text{thetas} \) 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 \( \text{Delta}_pq \).

\( q \) 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.

\( \text{spec} \) 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**

```r
data <- matrix(c(3,4,3,3,2,4,2,3,3,4,4,3,2,5,3,4,4,3,3,3,4,4,3,3,4,3,5,4,3,3,3), 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, 7)
MF112 <- TrapezoidalFuzzyNumber(2, 7, 15)
MF113 <- TrapezoidalFuzzyNumber(7, 15, 23)
MF114 <- TrapezoidalFuzzyNumber(15, 23, 28)
MF115 <- TrapezoidalFuzzyNumber(23, 28, 30)
MF11 <- GFUZZ(data, 1, 1, PA11, spec="Identical", breakpoints = 100)
# ------------------
MF121 <- TrapezoidalFuzzyNumber(0, 2, 7)
MF122 <- TrapezoidalFuzzyNumber(2, 7, 15)
MF123 <- TrapezoidalFuzzyNumber(7, 15, 23)
MF124 <- TrapezoidalFuzzyNumber(15, 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

Numerical integration by the composite trapezoidal method - method 3

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

Numerical integration by the Simpson method - method 4

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**

```r
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**

```r
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

Description

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

Description

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 ≤ q ≤ r ≤ s. No NA are allowed.
Kurtosis

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

Examples
```r
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
```r
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
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
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

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.
theta 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(1,2,2,3,3,4,4,5), 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

nbreakpoints(data)

Arguments

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)
alpha.X <- alphacut(X, seq(0,1,0.01))
nbreakpoints(alpha.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:

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

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**

\[ n_{jkq.}(x, \text{varindex}, q, \text{p\_ind} = \text{rep}(1, \text{nrow}(x))) \]
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 \( \text{length}(p_{\text{ind}}) = nrow(data) \). If the weights are not relative, the following expression should be applied on the vector:

\[
p_{\text{ind}} \sum_{i=1}^{N} p_{\text{ind}}.
\]

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

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/rddev.
- **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.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**

```r
p.value.log(
    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**

- **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.
**p.value.mean.log**

This function 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,  # Data fuzzified
  type,            # Type of test
  H0,              # Null hypothesis
  H1,              # Alternative hypothesis
  s.d,             # Standard deviation
  sig,             # Significance level
  dist.type,       # Distribution type
  i = 1,           # Parameter i
  j = 1,           # Parameter j
  theta = 1/3,     # Parameter theta
  thetas = 1,      # Parameter thetas
  p = 2,           # Parameter p
  q = 0.5,         # Parameter q
  breakpoints = 100  # Number of breakpoints
)
```

### Parameters

- **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.

---

**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,  # Data fuzzified
  type,            # Type of test
  H0,              # Null hypothesis
  H1,              # Alternative hypothesis
  s.d,             # Standard deviation
  sig,             # Significance level
  dist.type,       # Distribution type
  i = 1,           # Parameter i
  j = 1,           # Parameter j
  theta = 1/3,     # Parameter theta
  thetas = 1,      # Parameter thetas
  p = 2,           # Parameter p
  q = 0.5,         # Parameter q
  breakpoints = 100  # Number of breakpoints
)
```
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.
theta 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.

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.
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.
\texttt{p.value.mean.poisson}

\texttt{p.value.mean.poisson}

\texttt{p.value.mean.poisson}

\begin{itemize}
\item \texttt{p} \hspace{1cm} a positive integer such that \(1 \leq p < \infty\), referring to the parameter of the \texttt{Rho\_p} and \texttt{Delta\_pq}. By default, \(p\) is fixed to 2.
\item \texttt{q} \hspace{1cm} a decimal value between 0 and 1, referring to the parameter of the metric \texttt{Delta\_pq}. By default, \(p\) is fixed to 0.5.
\item \texttt{breakpoints} \hspace{1cm} a positive arbitrary integer representing the number of breaks chosen to build the numerical alpha-cuts. It is fixed to 100 by default.
\end{itemize}

\textbf{Value}

Returns the defuzzified p-value and the decision made.

---

\texttt{p.value.mean.poisson} \hspace{1cm} Calculates the p-value of fuzzy observations taken from a Poisson distribution for the mean

---

\textbf{Description}

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

\textbf{Usage}

\begin{verbatim}
p.value.mean.poisson(
data.fuzzified,
type,
H0,
H1,
sig,
dist.type,
i = 1,
j = 1,
theta = 1/3,
theetas = 1,
p = 2,
q = 0.5,
breakpoints = 100
)
\end{verbatim}

\textbf{Arguments}

\begin{itemize}
\item \texttt{data.fuzzified} \hspace{1cm} a fuzzification matrix constructed by a call to the function \texttt{FUZZ} or the function \texttt{GFUZZ}, or a similar matrix. No NA are allowed.
\item \texttt{type} \hspace{1cm} 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.
\item \texttt{H0} \hspace{1cm} a trapezoidal or a triangular fuzzy number representing the fuzzy null hypothesis.
\end{itemize}
p.value.mean.Student

- **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/denv/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** *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,
```
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 $1 \leq p < \infty$, referring to the parameter of the Rho_p and Delta_pq. By default, p is fixed to 1. This parameter is used in the calculations of the d_theta star and the d_GSGD distances.
- **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**

```r
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**

- **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", "Bertoluzzza", "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.
`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 
)
```

- **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.
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", "Rhoph", "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 $d_{theta}$ 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.

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

Value

Returns the defuzzified p-value and the decision made.
p.value.Student  \hspace{1cm} \textit{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

\begin{verbatim}
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
)
\end{verbatim}

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: \( d_{\text{Bertoluzza}}, d_{\text{mid/spr}} \) and \( d_{\text{phi-wabl/ldev/rdev}} \).

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

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

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

\( \text{s.d} \) 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

\[ R(x, \ p_\text{ind}, b\_j\_k, SI) \]

### Arguments

- \( x \) the data set to evaluate.
- \( p_\text{ind} \) a vector of the relative sampling weights of the units, for which \( \text{length}(p_\text{ind}) = \text{nrow}(data) \). If the weights are not relative, the following expression should be applied on the vector:

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

If no sampling weights are used, the vector of weights is reduced to a vector of values 1, i.e. \( \text{rep}(1, \text{nrow}(data)) \).
- \( b\_j\_k \) a matrix of \( \text{length}(b\_j) \) rows and \( \text{max}(SI) \) 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. 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

```r
data <- matrix(c(3,4,2,3,3,2,4,3,3,4,4,2,5,3,4,4,3,3,4,4,3,3,4,4,3, 3,4,3,3,3,4,3,5,3,4,3,3,3), ncol = 4)
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

```r
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 $Rho_p$ and $Delta_{pq}$.
- 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 <- matrix(c(3,4,2,3,2,4,3,3,4,3,4,4,2,5,3,4,4,3,3,3,4,4,3,3,3,3,4,3,3,3,4,3,3,3,4,3,3,3,4,3,3,3), ncol = 4)
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_Bertoluzza, d_mid/spr and d_phi-wabl/ldev/rdev.
theta*  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.
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

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

---

**square**

*Square a number*

Description

Takes any numerical value and squares it.

Usage

```r
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

```r
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.
Weighted.fuzzy.mean  Calculates the weighted fuzzy sample mean

Description
Calculates the weighted fuzzy sample mean

Usage
Weighted.fuzzy.mean(
  data.fuzzified,
  weight,
  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.
  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
  mat <- matrix(c(1,2,3,4,5), ncol =4)
  w <- c(1,3)
  Weighted.fuzzy.mean(mat, w)
Index

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
fci.ml, 22
fci.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
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
INDEX

<table>
<thead>
<tr>
<th>Term</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>is.trfuzzification</td>
<td>68</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>69</td>
</tr>
<tr>
<td>Mid.Spr</td>
<td>70</td>
</tr>
<tr>
<td>Moment</td>
<td>71</td>
</tr>
<tr>
<td>n_jk</td>
<td>73</td>
</tr>
<tr>
<td>n_jkq</td>
<td>73</td>
</tr>
<tr>
<td>nbreakpoints</td>
<td>72</td>
</tr>
<tr>
<td>p.value.fisher</td>
<td>74</td>
</tr>
<tr>
<td>p.value.log</td>
<td>76</td>
</tr>
<tr>
<td>p.value.mean.log</td>
<td>77</td>
</tr>
<tr>
<td>p.value.mean.normal</td>
<td>78</td>
</tr>
<tr>
<td>p.value.mean.poisson</td>
<td>80</td>
</tr>
<tr>
<td>p.value.mean.Student</td>
<td>81</td>
</tr>
<tr>
<td>p.value.normal</td>
<td>83</td>
</tr>
<tr>
<td>p.value.poisson</td>
<td>84</td>
</tr>
<tr>
<td>p.value.Student</td>
<td>86</td>
</tr>
<tr>
<td>R</td>
<td>87</td>
</tr>
<tr>
<td>Rho1</td>
<td>88</td>
</tr>
<tr>
<td>Rho2</td>
<td>89</td>
</tr>
<tr>
<td>Rhop</td>
<td>89</td>
</tr>
<tr>
<td>Ri</td>
<td>90</td>
</tr>
<tr>
<td>Sample.variance</td>
<td>91</td>
</tr>
<tr>
<td>SEQ.ORDERING</td>
<td>92</td>
</tr>
<tr>
<td>SEQ.ORDERING.APPROXIMATION</td>
<td>92</td>
</tr>
<tr>
<td>SEQ.ORDERING.EXACT</td>
<td>93</td>
</tr>
<tr>
<td>SGD</td>
<td>93</td>
</tr>
<tr>
<td>Skewness</td>
<td>94</td>
</tr>
<tr>
<td>square</td>
<td>95</td>
</tr>
<tr>
<td>tr.gfuzz</td>
<td>95</td>
</tr>
<tr>
<td>wabl</td>
<td>96</td>
</tr>
<tr>
<td>Weighted.fuzzy.mean</td>
<td>97</td>
</tr>
</tbody>
</table>