Package 'FuzzyResampling'

October 4, 2024

Type Package

Title Resampling Methods for Triangular and Trapezoidal Fuzzy Numbers

Version 0.6.4

Maintainer Maciej Romaniuk `<mroman@ibspan.waw.pl>`

Description

The classical (i.e. Efron's, see Efron and Tibshirani (1994, ISBN:978-0412042317) ``An Introduction to the Bootstrap") bootstrap is widely used for both the real (i.e. ``crisp") and fuzzy data. The main aim of the algorithms implemented in this package is to overcome a problem with repetition of a few distinct values and to create fuzzy numbers, which are ``similar" (but not the same) to values from the initial sample.

To do this, different characteristics of triangular/trapezoidal numbers are kept (like the value, the ambiguity, etc., see Grzegorzewski et al. <doi:10.2991/eusflat-19.2019.68>, Grzegorzewski et al. (2020) <doi:10.2991/ijcis.d.201012.003>, Grzegorzewski et al. (2020) <doi:10.34768/amcs-2020-0022>, Grzegorzewski and Romaniuk (2022) <doi:10.1007/978-3-030-95929-6_3>, Romaniuk and Hryniewicz (2019) <doi:10.1007/s00500-018-3251-5>).

Some additional procedures related to these resampling methods are also provided, like calculation of the Bertoluzza et al.'s distance (aka the mid/spread distance, see Bertoluzza et al. (1995) ``On a new class of distances between fuzzy numbers") and estimation of the p-value of the one- and two- sample bootstrapped test for the mean (see Lubiano et al. (2016, <doi:10.1016/j.ejor.2015.11.016>)).

Additionally, there are procedures which randomly generate trapezoidal fuzzy numbers using some well-known statistical distributions.

Imports stats, utils

URL https://github.com/mroman-ibs/FuzzyResampling

BugReports https://github.com/mroman-ibs/FuzzyResampling/issues

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0), R.rsp

Config/testthat/edition 3

VignetteBuilder R.rsp

NeedsCompilation no

Author Maciej Romaniuk [aut, cre],
       Przemyslaw Grzegorzewski [aut],
       Olgierd Hryniewicz [aut]

Repository CRAN

Date/Publication 2024-10-04 10:00:02 UTC

Contents

BertoluzzaDistance   *Calculate Bertoluzza's (mid/spread) distance for triangular and trape-*
                                zoidal fuzzy numbers

Description

BertoluzzaDistance returns the Bertoulzza et al.'s (aka mid/spread) distance with the given weight
theta between two triangular or trapezoidal fuzzy numbers.

Usage

```
BertoluzzaDistance(fuzzyNumber1, fuzzyNumber2, theta = 1/3, increases = FALSE)
```

Arguments

fuzzyNumber1   The first triangular or trapezoidal fuzzy number. More than one value can be given in the form of matrix.

fuzzyNumber2   The second triangular or trapezoidal fuzzy number. More than one value can be given in the form of matrix.

theta   The weighting parameter for the mid/spread distance.

increases   If TRUE is used, then the fuzzy numbers should be given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input fuzzy values should be triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

The procedure calculates the mid/spread distance between two fuzzy values based on the weight theta, usually we have theta = 1/3 or theta = 1. The output is given as vector of distances.

Value

This function returns vector of doubles related to the calculated distances.

References

Bertoluzza, C., Corral, N., Salas, A. (1995) On a new class of distances between fuzzy numbers Mathware and Soft Computing, 2 (2), pp. 71-84

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),ncol = 4,byrow = TRUE)

# calculate the mid/spread distance between the first value
# (from the first row) and the second one (from the second row)

BertoluzzaDistance(fuzzyValues[1,],fuzzyValues[2,])
```

```
# calculate the mid/spread distance between the first value
# (from the first row) and all of the values (from the first to the third row)

BertoluzzaDistance(fuzzyValues[1,],fuzzyValues)
```

CalculateAmbiguity   *Calculation of the ambiguity for triangular and trapezoidal fuzzy numbers*

Description

CalculateAmbiguity returns the ambiguity of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022)).

Usage

```
CalculateAmbiguity(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber	Input data consist of triangular or trapezoidal fuzzy numbers.
increases	If TRUE is used, then the initial data should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or
2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

Then for each fuzzy number, its characteristics, known as the ambiguity of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022).

Value

This function returns vector of double values. Each output value is equal to the ambiguity of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzzy Numbers: Approximations, Ranking and Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

CalculateFuzziness for calculation of the fuzziness, CalculateValue for calculation of the value, CalculateAmbiguityR for calculation of the right-hand ambiguity, CalculateAmbiguityL for calculation of the left-hand ambiguity, CalculateExpValue for calculation of the expected value, CalculateWidth for calculation of the width

Other characteristics of fuzzy numbers functions: CalculateAmbiguityL(), CalculateAmbiguityR(), CalculateExpValue(), CalculateFuzziness(), CalculateValue(), CalculateWidth()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the ambiguity of the first fuzzy number

CalculateAmbiguity(fuzzyValues[1,])

# calculate the ambiguity for the whole matrix

CalculateAmbiguity(fuzzyValues)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the ambiguity of the first fuzzy number

CalculateAmbiguity(fuzzyValuesInc[1,], increases = TRUE)
```

CalculateAmbiguityL	*Calculation of the left-hand ambiguity for triangular and trapezoidal fuzzy numbers*

Description

`CalculateAmbiguityL` returns the left-hand ambiguity of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022)).

Usage

```
CalculateAmbiguityL(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber   Input data consist of triangular or trapezoidal fuzzy numbers.

increases   If TRUE is used, then the initial data should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter `increases=TRUE` has to be used.

Then for each fuzzy number, its characteristics, known as the left-hand ambiguity of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022).

Value

This function returns vector of double values. Each output value is equal to the left-hand ambiguity of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzzy Numbers: Approximations, Ranking and Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

CalculateFuzziness for calculation of the fuzziness, CalculateValue for calculation of the value, CalculateAmbiguityR for calculation of the right-hand ambiguity, CalculateAmbiguity for calculation of the ambiguity, CalculateExpValue for calculation of the expected value, CalculateWidth for calculation of the width

Other characteristics of fuzzy numbers functions: CalculateAmbiguity(), CalculateAmbiguityR(), CalculateExpValue(), CalculateFuzziness(), CalculateValue(), CalculateWidth()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the left-hand ambiguity of the first fuzzy number

CalculateAmbiguityL(fuzzyValues[1,])

# calculate the left-hand ambiguity for the whole matrix

CalculateAmbiguityL(fuzzyValues)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the left-hand ambiguity of the first fuzzy number

CalculateAmbiguityL(fuzzyValuesInc[1,], increases = TRUE)
```

CalculateAmbiguityR	*Calculation of the right-hand ambiguity for triangular and trapezoidal fuzzy numbers*

Description

CalculateAmbiguityR returns the right-hand ambiguity of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022)).

Usage

```
CalculateAmbiguityR(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber   Input data consist of triangular or trapezoidal fuzzy numbers.

increases   If TRUE is used, then the initial data should consist of the fuzzy numbers in the
                form: left increment of the support, left end of the core, right end of the core,
                right increment of the support. Otherwise, the default value FALSE is used and
                the fuzzy numbers should be given in the form: left end of the support, left end
                of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or
a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the
   support.

In this second case, the parameter increases=TRUE has to be used.

Then for each fuzzy number, its characteristics, known as the right-hand ambiguity of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski and
Romaniuk, 2022).

Value

This function returns vector of double values. Each output value is equal to the right-hand ambiguity
of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzy Numbers: Approximations, Ranking and
Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives,
pp. 28-47 Springer

See Also

CalculateFuzziness for calculation of the fuzziness, CalculateValue for calculation of the
value, CalculateAmbiguityL for calculation of the left-hand ambiguity, CalculateAmbiguity for
calculation of the ambiguity, CalculateExpValue for calculation of the expected value, CalculateWidth
for calculation of the width

Other characteristics of fuzzy numbers functions: CalculateAmbiguity(), CalculateAmbiguityL(),
CalculateExpValue(), CalculateFuzziness(), CalculateValue(), CalculateWidth()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the right-hand ambiguity of the first fuzzy number

CalculateAmbiguityR(fuzzyValues[1,])

# calculate the right-hand ambiguity for the whole matrix

CalculateAmbiguityR(fuzzyValues)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the right-hand ambiguity of the first fuzzy number

CalculateAmbiguityR(fuzzyValuesInc[1,], increases = TRUE)
```

CalculateExpValue   *Calculation of the expected value for triangular and trapezoidal fuzzy numbers*

Description

CalculateExpValue returns the expected value of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022)).

Usage

```
CalculateExpValue(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber   Input data consist of triangular or trapezoidal fuzzy numbers.

increases   If TRUE is used, then the initial data should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

Then for each fuzzy number, its characteristics, known as the expected value of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022).

Value

This function returns vector of double values. Each output value is equal to the expected value of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzzy Numbers: Approximations, Ranking and Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

CalculateFuzziness for calculation of the fuzziness, CalculateAmbiguityL for calculation of the left-hand ambiguity, CalculateAmbiguityR for calculation of the right-hand ambiguity, CalculateAmbiguity for calculation of the ambiguity, CalculateWidth for calculation of the width, CalculateValue for calculation of the value

Other characteristics of fuzzy numbers functions: CalculateAmbiguity(), CalculateAmbiguityL(), CalculateAmbiguityR(), CalculateFuzziness(), CalculateValue(), CalculateWidth()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the expected value of the first fuzzy number

CalculateExpValue(fuzzyValues[1,])

# calculate the expected value for the whole matrix

CalculateExpValue(fuzzyValues)
```

```
# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the expected value of the first fuzzy number

CalculateExpValue(fuzzyValuesInc[1,], increases = TRUE)
```

CalculateFuzziness	*Calculation of the fuzziness for triangular and trapezoidal fuzzy numbers*

Description

CalculateFuzziness returns the fuzziness of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski et al., 2020)).

Usage

```
CalculateFuzziness(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber   Input data consist of triangular or trapezoidal fuzzy numbers.

increases   If TRUE is used, then the initial data should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

Then for each fuzzy number, its characteristics, known as the fuzziness of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski et al., 2020)).

Value

This function returns vector of double values. Each output value is equal to the fuzziness of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzy Numbers: Approximations, Ranking and Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data based on the canonical representation International Journal of Computational Intelligence Systems, 13 (1), pp. 1650-1662

See Also

CalculateValue for calculation of the value, CalculateAmbiguityL for calculation of the left-hand ambiguity, CalculateAmbiguityR for calculation of the right-hand ambiguity, CalculateAmbiguity for calculation of the ambiguity, CalculateExpValue for calculation of the expected value, CalculateWidth for calculation of the width

Other characteristics of fuzzy numbers functions: CalculateAmbiguity(), CalculateAmbiguityL(), CalculateAmbiguityR(), CalculateExpValue(), CalculateValue(), CalculateWidth()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the fuzziness of the first fuzzy number

CalculateFuzziness(fuzzyValues[1,])

# calculate the fuzziness for the whole matrix

CalculateFuzziness(fuzzyValues)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the fuzziness of the first fuzzy number

CalculateFuzziness(fuzzyValuesInc[1,], increases = TRUE)
```

| `CalculateValue` | *Calculation of the value for triangular and trapezoidal fuzzy numbers* |

Description

`CalculateValue` returns the value of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022)).

Usage

```
CalculateValue(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber   Input data consist of triangular or trapezoidal fuzzy numbers.

increases   If `TRUE` is used, then the initial data should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value `FALSE` is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or
2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter `increases=TRUE` has to be used.

Then for each fuzzy number, its characteristics, known as the value of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022).

Value

This function returns vector of double values. Each output value is equal to the value of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzzy Numbers: Approximations, Ranking and Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

CalculateFuzziness for calculation of the fuzziness, CalculateAmbiguityL for calculation of the left-hand ambiguity, CalculateAmbiguityR for calculation of the right-hand ambiguity, CalculateAmbiguity for calculation of the ambiguity, CalculateExpValue for calculation of the expected value, CalculateWidth for calculation of the width

Other characteristics of fuzzy numbers functions: CalculateAmbiguity(), CalculateAmbiguityL(), CalculateAmbiguityR(), CalculateExpValue(), CalculateFuzziness(), CalculateWidth()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the value of the first fuzzy number

CalculateValue(fuzzyValues[1,])

# calculate the values for the whole matrix

CalculateValue(fuzzyValues)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the value of the first fuzzy number

CalculateValue(fuzzyValuesInc[1,], increases = TRUE)
```

CalculateWidth   *Calculation of the width for triangular and trapezoidal fuzzy numbers*

Description

CalculateWidth returns the width of the triangular or trapezoidal fuzzy number (see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022)).

Usage

```
CalculateWidth(fuzzyNumber, increases = FALSE)
```

Arguments

fuzzyNumber	Input data consist of triangular or trapezoidal fuzzy numbers.
increases	If TRUE is used, then the initial data should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input data should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or
2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

Then for each fuzzy number, its characteristics, known as the width of fuzzy number, is calculated. For the respective formulas, see, e.g., (Ban et al., 2015), (Grzegorzewski and Romaniuk, 2022).

Value

This function returns vector of double values. Each output value is equal to the width of the respective fuzzy number.

References

Ban, A.I., Coroianu, L., Grzegorzewski, P. (2015) Fuzzy Numbers: Approximations, Ranking and Applications Institute of Computer Sciences, Polish Academy of Sciences

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

CalculateFuzziness for calculation of the fuzziness, CalculateAmbiguityL for calculation of the left-hand ambiguity, CalculateAmbiguityR for calculation of the right-hand ambiguity, CalculateAmbiguity for calculation of the ambiguity, CalculateExpValue for calculation of the expected value, CalculateValue for calculation of the value

Other characteristics of fuzzy numbers functions: CalculateAmbiguity(), CalculateAmbiguityL(), CalculateAmbiguityR(), CalculateExpValue(), CalculateFuzziness(), CalculateValue()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
```

```
ncol = 4,byrow = TRUE)

# calculate the width of the first fuzzy number

CalculateWidth(fuzzyValues[1,])

# calculate the width for the whole matrix

CalculateWidth(fuzzyValues)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# calculate the width of the first fuzzy number

CalculateWidth(fuzzyValuesInc[1,], increases = TRUE)
```

ClassicalBootstrap   *Classical bootstrap procedure for triangular and trapezoidal fuzzy*
                            numbers

Description

ClassicalBootstrap returns the bootstrapped (secondary) sample based on the initial sample and
uses the Efron's (i.e. classical) resampling scheme (see (Efron, 1994)).

Usage

```
ClassicalBootstrap(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample	Initial sample of triangular or trapezoidal fuzzy numbers.
b	The number of fuzzy values in the resampled (secondary) sample. If this parameter is not specified, the number of values in the initial sample is used. The parameter should be integer value more than 0.
increases	If TRUE is used, then the initial sample should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

The resampling procedure produces b fuzzy values, which are randomly chosen (with repetition) from the initial sample (without any alternations). If the parameter b is not specified, it is equal to the length of the initial sample. The output is given in the same form as the initial sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled fuzzy number. These fuzzy numbers have the same form as the values from the initial sample depending on the provided parameter increases.

References

Efron, B. (1994). An Introduction to the Bootstrap. CRC Press

See Also

VAMethod for the VA method, EWMethod for the EW method, VAFMethod for the VAF method, VAAMethod for the VAA method, DMethod for the d method, WMethod for the w method

Other resampling functions: DMethod(), EWMethod(), VAAMethod(), VAFMethod(), VAMethod(), WMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate bootstrap sample

set.seed(12345)

ClassicalBootstrap(fuzzyValues)

ClassicalBootstrap(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)
```

```
# generate bootstrap sample

ClassicalBootstrap(fuzzyValuesInc,increases = TRUE)

ClassicalBootstrap(fuzzyValuesInc,b=4,increases = TRUE)
```

ComparePowerOneSampleCTest

*Comparison of the resampling approaches based on the power for the
one-sample test for the mean.*

Description

ComparePowerOneSampleCTest returns the percentage of rejections for the one-sample C-test when
different resampling methods are used.

Usage

```
ComparePowerOneSampleCTest(
  generator,
  mu_0,
  shiftVector,
  sampleSize = 10,
  numberOfSamples = 10,
  initialSamples = 100,
  theta = 1/3,
  significance = 0.05,
  ...
)
```

Arguments

generator	Name of the generator for sampling initial samples. For the possible names check the values of samplingGenerators vector.
mu_0	Triangular or trapezoidal fuzzy number which is used for the null hypothesis of the C-test.
shiftVector	Deterministic vector of shifts that are sequentially added to all initial samples.
sampleSize	Size of the single initial sample.
numberOfSamples	
	Number of the bootstrapped samples used to estimate the p-value.
initialSamples	Number of the generated initial samples. More than one value can be given in the form of matrix.
theta	The weighting parameter for the mid/spread distance applied in the C-test.
significance	The significance value used to accept/reject the hypothesis for the one-sample C-test.
...	Parameters which are passed to OneSampleCTest or the respective generator

Details

The function generates a sequence of initial samples (their number is given in `initialSamples`, the size is determined by `sampleSize`) for fuzzy numbers of the type specified by `generator`. Then a sequence of deterministic shifts described by vector `shiftVector` is added to each fuzzy observation in these samples. Next, function `OneSampleCTest` is executed to calculate the p-value for each combination of the initial sample and resampling method. Then, by comparing the p-value with the assumed significance level `significance` we make a decision whether to reject the null hypothesis for the one-sample C-test for the mean (see Lubiano et al. (2016)) or not. The output of this procedure is the percentage of rejections as a function of values from `shiftVector`.

Value

This function returns a matrix of percentage of rejections for the one-sample C-test for the mean. Rows in this matrix are related to the values from `shiftVector`, and the columns - to all resampling methods.

References

Lubiano, M.A., Montenegro M., Sinova, B., de Saa, S.R., Gil, M.A. (2016) Hypothesis testing for means in connection with fuzzy rating scale-based data: algorithms and applications European Journal of Operational Research, 251, pp. 918-929

See Also

ComparisonSEMean for the comparison of resampling methods based on SE/MSE for the mean, ComparisonOneSampleCTest for the comparison of resampling methods based on power for the one-sample C-test for the mean.

Other comparison of resampling methods: ComparisonOneSampleCTest(), ComparisonSEMean()

Examples

```
## Not run:

# seed PRNG

set.seed(1234)

# compare the resampling methods for the synthetic data generated using GeneratorNU function
# and two values of the shifts

ComparePowerOneSampleCTest("GeneratorNU",mu_0 = c(-0.4,-0.1,0.1,0.4), shiftVector = c(0,0.5),
mu = 0, sigma = 1, a = 0.2, b = 0.6)
## End(Not run)
```

ComparisonOneSampleCTest

> *Comparison of the resampling approaches based on the power for the one-sample test for the mean.*

Description

ComparisonOneSampleCTest returns the percentage of rejections for the one-sample C-test when different resampling methods are used.

Usage

```
ComparisonOneSampleCTest(
  generator,
  mu_0,
  shift = 0,
  sampleSize = 10,
  numberOfSamples = 10,
  initialSamples = 100,
  theta = 1/3,
  significance = 0.05,
  ...
)
```

Arguments

generator	Name of the generator for sampling initial samples. For the possible names check the values of samplingGenerators vector.
mu_0	Triangular or trapezoidal fuzzy number which is used for the null hypothesis of the C-test.
shift	Deterministic shift added to all initial samples.
sampleSize	Size of the single initial sample.
numberOfSamples	
	Number of the bootstrapped samples used to estimate the p-value.
initialSamples	Number of the generated initial samples. More than one value can be given in the form of matrix.
theta	The weighting parameter for the mid/spread distance applied in the C-test.
significance	The significance value used to accept/reject the hypothesis for the one-sample C-test.
...	Parameters which are passed to OneSampleCTest or the respective generator

Details

The function generates a sequence of initial samples (their number is given in `initialSamples`, the size is determined by `sampleSize`) for fuzzy numbers of the type specified by `generator`. Then some deterministic shift of the size `shift` is added to each fuzzy observation in these samples. Next, function `OneSampleCTest` is executed to calculate the p-value for each combination of the initial sample and resampling method. Then, by comparing the p-value with the assumed significance level `significance` we make a decision whether to reject the null hypothesis for the one-sample C-test for the mean (see Lubiano et al. (2016)) or not. The output of this procedure is the percentage of rejections in the sequence of experiments.

Value

This function returns a vector of percentage of rejections for the one-sample C-test for the mean.

References

Lubiano, M.A., Montenegro M., Sinova, B., de Saa, S.R., Gil, M.A. (2016) Hypothesis testing for means in connection with fuzzy rating scale-based data: algorithms and applications European Journal of Operational Research, 251, pp. 918-929

See Also

ComparisonSEMean for the comparison of resampling methods based on SE/MSE for the mean, ComparePowerOneSampleCTest for the comparison of resampling methods based on power for the one-sample C-test for the mean.

Other comparison of resampling methods: ComparePowerOneSampleCTest(), ComparisonSEMean()

Examples

```
## Not run:

# seed PRNG

set.seed(1234)

# compare the resampling methods for the synthetic data generated using GeneratorNU function

ComparisonOneSampleCTest(generator="GeneratorNU",mu_0 = c(-0.4,-0.1,0.1,0.4),
 sampleSize = 10,numberOfSamples = 100, initialSamples = 100,mu = 0, sigma = 1,a = 0.2,b = 0.6)
## End(Not run)
```

ComparisonSEMean   *Comparison of the resampling approaches based on the SE/MSE for the mean.*

Description

ComparisonSEMean estimates the standard error (SE) or the mean-squared error (MSE) for the mean for all resampling approaches.

Usage

```
ComparisonSEMean(
  generator,
  sampleSize = 10,
  numberOfSamples = 100,
  repetitions = 100,
  trueMean = NA,
  theta = 1/3,
  ...
)
```

Arguments

generator	Name of the generator for sampling initial samples. For the possible names check the values of samplingGenerators vector.
sampleSize	Size of the single initial sample.
numberOfSamples	
	Number of the initial samples.
repetitions	Number of the secondary samples which are created using the selected resampling method.
trueMean	If the value is given, then the mean-squared error (MSE) is calculated for this value and the means of the bootstrapped samples. Otherwise, the standard error (SE) is calculated based on the overall mean of the secondary samples.
theta	The weighting parameter for the mid/spread distance applied in the C-test.
...	Parameters which are passed to SEResamplingMean or the respective generator

Details

The function generates a sequence of initial samples (their number is given in initialSamples, the size is determined by sampleSize) for fuzzy numbers of the type specified by generator. Then the SE/MSE is calculated for each combination of the initial sample and resampling method using SEResamplingMean. The output values are the SE/MSE averaged by initialSamples.

Value

This function returns a vector of the averaged SE/MSE for the mean.

References

Bertoluzza, C., Corral, N., Salas, A. (1995) On a new class of distances between fuzzy numbers Mathware and Soft Computing, 2 (2), pp. 71-84

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

ComparisonOneSampleCTest for the comparison of resampling methods based on power for the one-sample C-test for the mean, ComparePowerOneSampleCTest for the comparison of resampling methods based on power for the one-sample C-test for the mean.

Other comparison of resampling methods: ComparePowerOneSampleCTest(), ComparisonOneSampleCTest()

Examples

```
## Not run:

# seed PRNG

set.seed(1234)

# calculate the SE of the mean for the synthetic data generated using GeneratorNU function

ComparisonSEMean(generator = "GeneratorNU",sampleSize = 10,
 numberOfSamples = 100, repetitions = 10,mu = 0, sigma = 1,a = 0.5, b = 1)
## End(Not run)
```

DMethod   *d method for resampling triangular and trapezoidal fuzzy numbers*

Description

DMethod returns the secondary (bootstrapped) sample and uses the resampling scheme which does not change the left end of the cores and increments (i.e. length of the core, left and right increment of the support) of the fuzzy variables from the initial sample (the d method, see (Romaniuk and Hryniewicz, 2019)).

Usage

```
DMethod(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample	Initial sample of triangular or trapezoidal fuzzy numbers.
b	The number of fuzzy values in the resampled (secondary) sample. If this parameter is not specified, the number of values in the initial sample is used. The parameter should be integer value more than 0.
increases	If TRUE is used, then the initial sample should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

The resampling procedure produces b fuzzy values. During the first step, the four values are randomly chosen based on the whole initial sample: left end of the core, length of the core, left and right increment of the support (with equal probabilities for each fuzzy variable). Then the new fuzzy variable, which preserves these characteristics, is created. If the parameter b is not specified, it is equal to the length of the initial sample. The output is given in the same form as the initial sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled fuzzy number. These fuzzy numbers have the same form as the values from the initial sample depending on the provided parameter increases.

References

Romaniuk, M., Hryniewicz, O. (2019) Interval-based, nonparametric approach for resampling of fuzzy numbers Soft Computing, 23 (14), pp. 5883-5903

See Also

ClassicalBootstrap, EWMethod for the EW method, VAFMethod for the VAF method, VAAMethod for the VAA method, WMethod for the w method

Other resampling functions: ClassicalBootstrap(), EWMethod(), VAAMethod(), VAFMethod(), VAMethod(), WMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the d method

set.seed(12345)

DMethod(fuzzyValues)

DMethod(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the d method

DMethod(fuzzyValuesInc,increases = TRUE)

DMethod(fuzzyValuesInc,b=4,increases = TRUE)
```

EWMethod	*E(xpected value)W(idth) resampling method for triangular and trape-zoidal fuzzy numbers*

Description

EWMethod returns the secondary (bootstrapped) sample and uses the resampling scheme which does not change the expected values and widths of the fuzzy variables from the initial sample (the EW method, see (Grzegorzewski et al, 2020)).

Usage

```
EWMethod(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample   Initial sample of triangular or trapezoidal fuzzy numbers.

b   The number of fuzzy values in the resampled (secondary) sample. If this parameter is not specified, the number of values in the initial sample is used. The parameter should be integer value more than 0.

increases   If `TRUE` is used, then the initial sample should consist of the fuzzy numbers in
                   the form: left increment of the support, left end of the core, right end of the core,
                   right increment of the support. Otherwise, the default value `FALSE` is used and
                   the fuzzy numbers should be given in the form: left end of the support, left end
                   of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector
or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the
   support.

In this second case, the parameter `increases=TRUE` has to be used.

The resampling procedure produces b fuzzy values. During the first step, the fuzzy value from the
initial sample is randomly chosen (with repetition). Then the new fuzzy variable, which preserves
the expected value and width of the old one, is randomly created. If the parameter b is not specified,
it is equal to the length of the initial sample. The output is given in the same form as the initial
sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled
fuzzy number. These fuzzy numbers have the same form as the values from the initial sample
depending on the provided parameter `increases`.

References

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data International Journal of Applied Mathematics and Computer Science, 30 (2), pp. 281-297

See Also

ClassicalBootstrap, VAMethod for the VA method, VAFMethod for the VAF method, VAAMethod
for the VAA method, DMethod for the d method, WMethod for the w method

Other resampling functions: ClassicalBootstrap(), DMethod(), VAAMethod(), VAFMethod(),
VAMethod(), WMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the EW method

set.seed(12345)
```

```
EWMethod(fuzzyValues)

EWMethod(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the EW method

EWMethod(fuzzyValuesInc,increases = TRUE)

EWMethod(fuzzyValuesInc,b=4,increases = TRUE)
```

GeneratorFuzzyNumbers   *Generate initial sample using various random distributions.*

Description

GeneratorFuzzyNumbers generates the random sample of trapezoidal fuzzy numbers using the various random distributions, using the functions from the stats package.

Usage

```
GeneratorFuzzyNumbers(
  n,
  originalRandomDist,
  parametersOriginalRD,
  increasesRandomDist,
  parametersIncreasesRD,
  supportLeftRandomDist,
  parametersSupportLeftRD,
  supportRightRandomDist,
  parametersSupportRightRD,
  increases = FALSE,
  ...
)
```

Arguments

n   Number of fuzzy trapezoidal numbers in the created sample.

originalRandomDist

                Name of the random generator used to create the "true origin" of fuzzy trape-
                zoidal number (as defined in stats package).

parametersOriginalRD

> List of parameters required by the random generator used to create the "true origin" of fuzzy trapezoidal number.

increasesRandomDist

> Name of the random generator used to create the increases of the core of fuzzy trapezoidal number (as defined in `stats` package).

parametersIncreasesRD

> List of parameters required by the random generator used to create the increases of the core of fuzzy trapezoidal number.

supportLeftRandomDist

> Name of the random generator used to create the increases of the left support of fuzzy trapezoidal number (as defined in `stats` package).

parametersSupportLeftRD

> List of parameters required by the random generator used to create the increases of the left support of fuzzy trapezoidal number.

supportRightRandomDist

> Name of the random generator used to create the increases of the right support of fuzzy trapezoidal number (as defined in `stats` package).

parametersSupportRightRD

> List of parameters required by the random generator used to create the increases of the right support of fuzzy trapezoidal number.

increases   If `TRUE` is used, then the output fuzzy numbers are given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value `FALSE` is used and the fuzzy numbers are given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

...   Some additional parameters to pass to other functions.

Details

The procedure simulates the initial sample which consists of n trapezoidal fuzzy numbers. The "true origin" of each fuzzy number is independently drawn from the random distribution using `originalRandomDist` function from the `stats` package with the parameters defined by `parametersOriginalRD`. The same applies to the increases of the core (the function `increasesRandomDist` with the parameters `parametersIncreasesRD` is then used), the left increase of the support (the function `supportLeftRandomDist` with the parameters `parametersSupportLeftRD`, respectively), and the right increase of the support (the function `supportRightRandomDist` with the parameters `parametersSupportRightRD`, respectively).

Names of these generators for random probabilities and their respective parameters should be in the form required by `stats` package.

The output is given as a matrix. In each row, there is a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

To obtain this second form, the parameter `increases=TRUE` has to be used.

Value

This function returns matrix which consists of n trapezoidal fuzzy numbers. Each fuzzy number (four values) ia given as one row in this matrix.

References

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data based on the canonical representation International Journal of Computational Intelligence Systems, 13 (1), pp. 1650-1662

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrapped Kolmogorov-Smirnov Test for Epistemic Fuzzy Data Information Processing and Management of Uncertainty in Knowledge-Based Systems Springer

See Also

GeneratorNExpUU

Other sampling functions: GeneratorNExpUU(), GeneratorNU()

Examples

```
# seed PRNG

set.seed(1234)

# generate 10 trapezoidal fuzzy numbers (the normal distribution for the "true origin",
# the exponential distribution for the increases of the core, and two different uniform
# distribution for the increases of the support)

GeneratorFuzzyNumbers(10,"rnorm",list(mean=0,sd=1),"rexp",list(rate=2),"runif",
                      list(min=0,max=0.6),"runif",list(min=0,max=1))
```

GeneratorNExpUU   *Generate initial sample using the normal and uniform distributions.*

Description

GeneratorNExpUU generates the random sample of trapezoidal fuzzy numbers using the normal, exponential and uniform distributions (for the "true" origin of each fuzzy number, and its increases, respectively).

Usage

```
GeneratorNExpUU(n, mu, sigma, lambda, b, c, increases = FALSE, ...)
```

Arguments

n	Number of fuzzy trapezoidal numbers in the created sample.
mu	The expected value of the normal distribution which is used to create the "true origin" of fuzzy trapezoidal number.
sigma	The standard deviation of the normal distribution which is used to create the "true origin" of fuzzy trapezoidal number.
lambda	The parameter of the exponential distributions which are used to create the left and right increases of the core.
b	The right end of the interval of the independent uniform distribution which is used to create the left increase of the support.
c	The right end of the interval of the independent uniform distribution which is used to create the right increase of the support.
increases	If TRUE is used, then the output fuzzy numbers are given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers are given in the form: left end of the support, left end of the core, right end of the core, right end of the support.
...	Some additional parameters to pass to other functions.

Details

The procedure simulates the initial sample which consists of n trapezoidal fuzzy numbers. The "true origin" of each fuzzy number is independently drawn from the normal distribution N (mu, sigma). Then increases of its core are independently generated from the exponential distributions Exp (lambda). The increases of its support are independently drawn from the uniform distributions U [0,b] (the left-hand increase) and U [0,c] (its right-hand counterpart) (see Grzegorzewski et al. (2020)).

The output is given as a matrix. In each row, there is a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

To obtain this second form, the parameter increases=TRUE has to be used.

Value

This function returns matrix which consists of n trapezoidal fuzzy numbers. Each fuzzy number (four values) ia given as one row in this matrix.

References

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data based on the canonical representation International Journal of Computational Intelligence Systems, 13 (1), pp. 1650-1662

See Also

GeneratorNU, GeneratorFuzzyNumbers

Other sampling functions: GeneratorFuzzyNumbers(), GeneratorNU()

Examples

```
# seed PRNG

set.seed(1234)

# generate 10 trapezoidal fuzzy numbers

GeneratorNExpUU(10, 0,1,1,1,2)
```

| GeneratorNU | *Generate initial sample using the normal and uniform distributions.* |

Description

GeneratorNU generates the random sample of trapezoidal fuzzy numbers using the normal and uniform distributions (for the "true" origin of each fuzzy number, and its increases, respectively).

Usage

```
GeneratorNU(n, mu, sigma, a, b, increases = FALSE, ...)
```

Arguments

n	Number of fuzzy trapezoidal numbers in the created sample.
mu	The expected value of the normal distribution which is used to create the "true origin" of fuzzy trapezoidal number.
sigma	The standard deviation of the normal distribution which is used to create the "true origin" of fuzzy trapezoidal number.
a	The right end of the interval of the independent uniform distributions which are used to create the left and right increases of the core.
b	The right end of the interval of the independent uniform distributions which are used to create the left and right increases of the support.
increases	If TRUE is used, then the output fuzzy numbers are given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers are given in the form: left end of the support, left end of the core, right end of the core, right end of the support.
...	Some additional parameters to pass to other functions.

Details

The procedure simulates the initial sample which consists of n trapezoidal fuzzy numbers. The "true origin" of each fuzzy number is independently drawn from the normal distribution N (mu, sigma). Then increases of its core and support are independently generated from the uniform distributions U [0,a] and U [0,b] (see Grzegorzewski et al. (2020)).

The output is given as a matrix. In each row, there is a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

To obtain this second form, the parameter increases=TRUE has to be used.

Value

This function returns matrix which consists of n trapezoidal fuzzy numbers. Each fuzzy number (four values) ia given as one row in this matrix.

References

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data based on the canonical representation International Journal of Computational Intelligence Systems, 13 (1), pp. 1650-1662

See Also

GeneratorNExpUU, GeneratorFuzzyNumbers

Other sampling functions: GeneratorFuzzyNumbers(), GeneratorNExpUU()

Examples

```
# seed PRNG

set.seed(1234)

# generate 10 trapezoidal fuzzy numbers

GeneratorNU(10, 0,1,1,2)
```

OneSampleCTest   *Calculate p-value of the one-sample test for the mean*

Description

OneSampleCTest returns the p-value of the one-sample test for the mean using the resampling method.

Usage

```
OneSampleCTest(
  initialSample,
  mu_0,
  numberOfSamples = 100,
  theta = 1/3,
  resamplingMethod = "ClassicalBootstrap",
  increases = FALSE
)
```

Arguments

initialSample	The initial sample which consists of triangular or trapezoidal fuzzy numbers. More than one value can be given in the form of matrix.
mu_0	Triangular or trapezoidal fuzzy number which is used for the null hypothesis of the C-test.
numberOfSamples	
	Number of the bootstrapped samples used to estimate the p-value.
theta	The weighting parameter for the mid/spread distance applied in the C-test.
resamplingMethod	
	Name of the resampling method, which is used to generate the bootstrapped samples. For the possible names check the values of resamplingMethods vector.
increases	If TRUE is used, then the fuzzy numbers should be given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input fuzzy values should be triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

The procedure uses the resampling method given in the resamplingMethod parameter to estimate the p-value of the one-sample test for the mean (denoted further as the one-sample C-test, see Lubiano et al. (2016)). This test checks the null hypothesis that the Aumann-type mean of the fuzzy numbers is equal to a given fuzzy number mu_0.

Value

This function returns double value which is equal to the p-value of the one-sample C-test.

References

Lubiano, M.A., Montenegro M., Sinova, B., de Saa, S.R., Gil, M.A. (2016) Hypothesis testing for means in connection with fuzzy rating scale-based data: algorithms and applications European Journal of Operational Research, 251, pp. 918-929

See Also

TwoSampleCTest for the two-sample C-test

Other bootstrapped version of test: TwoSampleCTest()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),ncol = 4,byrow = TRUE)

# seed PRNG

set.seed(1234)

# calculate the p-value using the classical (i.e. Efron's) bootstrap

OneSampleCTest(fuzzyValues, mu_0 = c(0,0.5,1,1.5))

# calculate the p-value using the VA resampling method

OneSampleCTest(fuzzyValues, mu_0 = c(0,0.5,1,1.5),resamplingMethod = "VAMethod")
```

---

resamplingMethods   *A vector containing names of all resampling methods.*

Description

resamplingMethods is a vector containing names of all resampling methods.

Usage

```
resamplingMethods
```

Format

An object of class `character` of length 7.

Value

This function returns a vector of strings.

Examples

```
# check the names of the available resampling methods

resamplingMethods
```

---

samplingGenerators   *A vector containing names of all sampling generators*

Description

`samplingGenerators` is a vector containing names of all generators which can be used to sample synthetic fuzzy numbers (trapezoidal or triangular).

Usage

```
samplingGenerators
```

Format

An object of class `character` of length 3.

Value

This function returns a vector of strings.

Examples

```
# check the names of the available sampling generators

samplingGenerators
```

SEResamplingMean   *Calculate SE/MSE for the mean of the bootstrapped samples.*

Description

SEResamplingMean estimates the standard error (SE) or the mean-squared error (MSE) for the mean while the sample is bootstrapped using one of the applied resampling methods.

Usage

```
SEResamplingMean(
  initialSample,
  resamplingMethod = "ClassicalBootstrap",
  repetitions = 100,
  trueMean = NA,
  theta = 1/3,
  increases = FALSE
)
```

Arguments

initialSample   The initial sample which consists of triangular or trapezoidal fuzzy numbers. More than one value can be given in the form of matrix.

resamplingMethod

Name of the resampling method, which is used to generate the bootstrapped samples. For the possible names check the values of resamplingMethods vector.

repetitions   Number of the secondary samples which are created using the selected resampling method.

trueMean   If the value is given, then the mean-squared error (MSE) is calculated for this value and the means of the bootstrapped samples. Otherwise, the standard error (SE) is calculated based on the overall mean of the secondary samples.

theta   The weighting parameter for the mid/spread distance applied to calculate the SE/MSE.

increases   If TRUE is used, then the fuzzy numbers should be given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input fuzzy values should be triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter `increases=TRUE` has to be used.

The procedure uses the resampling method given in `resamplingMethod` parameter to create the secondary (bootstrapped) samples. Then the mean (in the form of trapezoidal fuzzy number) is calculated for each new sample. If the parameter `trueMean` is set, then the mean-squared error (MSE) between this true value and each of the sample means is estimated using the Bertoulzza et al.'s (aka mid/spread) distance (with the given weight `theta`, see Bertoluzza et al. (1995)). Otherwise, the overall mean is calculated, and the standard error (SE) is estimated based on the overall mean and each of the sample means. In this case the Bertoulzza et al.'s distance (see Grzegorzewski, Romaniuk (2021)) is also applied.

Value

This function returns list of two double values:

1. mean which is equal to the overall mean (if the SE is calculated) or `trueMean` parameter (if the MSE is calculated),

2. SE which is equal to the estimated SE/MSE of the mean.

The output `mean` consists of four values which create the trapezoidal fuzzy number.

References

Bertoluzza, C., Corral, N., Salas, A. (1995) On a new class of distances between fuzzy numbers Mathware and Soft Computing, 2 (2), pp. 71-84

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),ncol = 4,byrow = TRUE)

# seed PRNG

set.seed(1234)

# calculate the SE of the mean using the classical (i.e. Efron's) bootstrap

SEResamplingMean(fuzzyValues)

# calculate the SE of the mean using the VA resampling method

SEResamplingMean(fuzzyValues, resamplingMethod = "VAMethod")

# calculate the MSE of the given mean using the classical (i.e. Efron's) bootstrap
```

```
SEResamplingMean(fuzzyValues, trueMean = c(0,0.5,1,2))

# calculate the MSE of the given mean using the VA resampling method

SEResamplingMean(fuzzyValues, resamplingMethod = "VAMethod", trueMean = c(0,0.5,1,2))
```

TwoSampleCTest   *Calculate p-value of the two-sample test for the mean*

Description

TwoSampleCTest returns the p-value of the two-sample test for the mean using the resampling method.

Usage

```
TwoSampleCTest(
  initialSample1,
  initialSample2,
  numberOfSamples = 100,
  theta = 1/3,
  resamplingMethod = "ClassicalBootstrap",
  increases = FALSE
)
```

Arguments

initialSample1   The first initial sample which consists of triangular or trapezoidal fuzzy numbers. More than one value can be given in the form of matrix.

initialSample2   The second initial sample which consists of triangular or trapezoidal fuzzy numbers. More than one value can be given in the form of matrix.

numberOfSamples
                  Number of the bootstrapped samples used to estimate the p-value.

theta   The weighting parameter for the mid/spread distance applied in the C-test.

resamplingMethod
                  Name of the resampling method, which is used to generate the bootstrapped samples. For the possible names check the values of resamplingMethods vector.

increases   If TRUE is used, then the fuzzy numbers should be given in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The input fuzzy values should be triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or
2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

The procedure uses the resampling method given in the resamplingMethod parameter to estimate the p-value of the two-sample test for the mean (denoted further as the two-sample C-test, see Lubiano et al. (2016)). This test checks the null hypothesis that the Aumann-type means of two fuzzy samples are equal.

Value

This function returns double value which is equal to the p-value of the two-sample C-test.

References

Lubiano, M.A., Montenegro M., Sinova, B., de Saa, S.R., Gil, M.A. (2016) Hypothesis testing for means in connection with fuzzy rating scale-based data: algorithms and applications European Journal of Operational Research, 251, pp. 918-929

See Also

OneSampleCTest for the one-sample C-test

Other bootstrapped version of test: OneSampleCTest()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),ncol = 4,byrow = TRUE)

# prepare the slightly shifted second sample

fuzzyValuesShift <- fuzzyValues + 0.5

# seed PRNG

set.seed(1234)

# calculate the p-value using the classical (i.e. Efron's) bootstrap

TwoSampleCTest(fuzzyValues, fuzzyValuesShift)

# calculate the p-value using the VA resampling method

TwoSampleCTest(fuzzyValues, fuzzyValuesShift, resamplingMethod = "VAMethod")
```

VAAMethod   *V(alue)A(mbiguity,   left-hand)A(mbiguity,   right-hand)   resampling*
                               method for triangular and trapezoidal fuzzy numbers

Description

`VAAMethod` returns the secondary (bootstrapped) sample and uses the resampling scheme which
does not change the values, left-hand and right-hand ambiguities of the fuzzy variables from the
initial sample (the VAA method, see (Grzegorzewski and Romaniuk, 2022)).

Usage

```
VAAMethod(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample	Initial sample of triangular or trapezoidal fuzzy numbers.
b	The number of fuzzy values in the resampled (secondary) sample. If this parameter is not specified, the number of values in the initial sample is used. The parameter should be integer value more than 0.
increases	If `TRUE` is used, then the initial sample should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value `FALSE` is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector
or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the
   support.

In this second case, the parameter `increases=TRUE` has to be used.

The resampling procedure produces b fuzzy values. During the first step, the fuzzy value from the
initial sample is randomly chosen (with repetition). Then the new fuzzy variable, which preserves
the value, left- and right-hand ambiguities of the old one, is randomly created. If the parameter b is
not specified, it is equal to the length of the initial sample. The output is given in the same form as
the initial sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled fuzzy number. These fuzzy numbers have the same form as the values from the initial sample depending on the provided parameter `increases`.

References

Grzegorzewski, P., Romaniuk, M. (2022) Bootstrap methods for fuzzy data Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, pp. 28-47 Springer

See Also

ClassicalBootstrap, EWMethod for the EW method, VAFMethod for the VAF method, DMethod for the d method, WMethod for the w method

Other resampling functions: ClassicalBootstrap(), DMethod(), EWMethod(), VAFMethod(), VAMethod(), WMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the VAA method

set.seed(12345)

VAAMethod(fuzzyValues)

VAAMethod(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the VAA method

VAAMethod(fuzzyValuesInc,increases = TRUE)

VAAMethod(fuzzyValuesInc,b=4,increases = TRUE)
```

VAFMethod   *V(alue)A(mbiguity)F(uzziness) resampling method for triangular and*
                                             trapezoidal fuzzy numbers

---

Description

VAFMethod returns the secondary (bootstrapped) sample and uses the resampling scheme which
does not change the values, ambiguities and fuzziness of the fuzzy variables from the initial sample
(the VAF method, see (Grzegorzewski et al., 2020)).

Usage

```
VAFMethod(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample	Initial sample of triangular or trapezoidal fuzzy numbers.
b	The number of fuzzy values in the resampled (secondary) sample. If this parameter is not specified, the number of values in the initial sample is used. The parameter should be integer value more than 0.
increases	If TRUE is used, then the initial sample should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector
or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the
   support.

In this second case, the parameter increases=TRUE has to be used.

The resampling procedure produces b fuzzy values. During the first step, the fuzzy value from the
initial sample is randomly chosen (with repetition). Then the new fuzzy variable, which preserves
the value, ambiguity and fuzziness of the old one, is randomly created. If the parameter b is not
specified, it is equal to the length of the initial sample. The output is given in the same form as the
initial sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled
fuzzy number. These fuzzy numbers have the same form as the values from the initial sample
depending on the provided parameter increases.

References

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data based on the canonical representation International Journal of Computational Intelligence Systems, 13 (1), pp. 1650-1662

See Also

ClassicalBootstrap, EWMethod for the EW method, VAFMethod for the VAF method, VAAMethod for the VAA method, DMethod for the d method, WMethod for the w method

Other resampling functions: ClassicalBootstrap(), DMethod(), EWMethod(), VAAMethod(), VAMethod(), WMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the VAF method

set.seed(12345)

VAFMethod(fuzzyValues)

VAFMethod(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the VAF method

VAFMethod(fuzzyValuesInc,increases = TRUE)

VAFMethod(fuzzyValuesInc,b=4,increases = TRUE)
```

VAMethod	*V(alue)A(mbiguity) resampling method for triangular and trapezoidal fuzzy numbers*

Description

VAMethod returns the secondary (bootstrapped) sample and uses the resampling scheme which does not change the values and ambiguities of the fuzzy variables from the initial sample (the VA method, see (Grzegorzewski et al, 2020)).

Usage

```
VAMethod(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample	Initial sample of triangular or trapezoidal fuzzy numbers.
b	The number of fuzzy values in the resampled (secondary) sample. If this parameter is not specified, the number of values in the initial sample is used. The parameter should be integer value more than 0.
increases	If TRUE is used, then the initial sample should consist of the fuzzy numbers in the form: left increment of the support, left end of the core, right end of the core, right increment of the support. Otherwise, the default value FALSE is used and the fuzzy numbers should be given in the form: left end of the support, left end of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the support.

In this second case, the parameter increases=TRUE has to be used.

The resampling procedure produces b fuzzy values. During the first step, the fuzzy value from the initial sample is randomly chosen (with repetition). Then the new fuzzy variable, which preserves the value and ambiguity of the old one, is randomly created. If the parameter b is not specified, it is equal to the length of the initial sample. The output is given in the same form as the initial sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled fuzzy number. These fuzzy numbers have the same form as the values from the initial sample depending on the provided parameter increases.

References

Grzegorzewski, P., Hryniewicz, O., Romaniuk, M. (2020) Flexible resampling for fuzzy data based on the canonical representation International Journal of Computational Intelligence Systems, 13 (1), pp. 1650-1662

See Also

ClassicalBootstrap, EWMethod for the EW method, VAFMethod for the VAF method, VAAMethod for the VAA method, DMethod for the d method, WMethod for the w method

Other resampling functions: ClassicalBootstrap(), DMethod(), EWMethod(), VAAMethod(), VAFMethod(), WMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the VA method

set.seed(12345)

VAMethod(fuzzyValues)

VAMethod(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the VA method

VAMethod(fuzzyValuesInc,increases = TRUE)

VAMethod(fuzzyValuesInc,b=4,increases = TRUE)
```

WMethod	*w method for resampling triangular and trapezoidal fuzzy numbers*

Description

`WMethod` returns the secondary (bootstrapped) sample and uses the resampling scheme based on the special *w density* which is related to the left ends of the cores and increments (i.e. length of the core, left and right increment of the support) of the fuzzy variables from the initial sample (the d method, see (Romaniuk and Hryniewicz, 2019)).

Usage

```
WMethod(initialSample, b = n, increases = FALSE)
```

Arguments

initialSample   Initial sample of triangular or trapezoidal fuzzy numbers.

b   The number of fuzzy values in the resampled (secondary) sample. If this pa-
                rameter is not specified, the number of values in the initial sample is used. The
                parameter should be integer value more than 0.

increases   If TRUE is used, then the initial sample should consist of the fuzzy numbers in
                     the form: left increment of the support, left end of the core, right end of the core,
                     right increment of the support. Otherwise, the default value FALSE is used and
                     the fuzzy numbers should be given in the form: left end of the support, left end
                     of the core, right end of the core, right end of the support.

Details

The initial sample should consist of triangular or trapezoidal fuzzy numbers, given as a single vector
or a whole matrix. In each row, there should be a single fuzzy number in one of the forms:

1. left end of the support, left end of the core, right end of the core, right end of the support, or

2. left increment of the support, left end of the core, right end of the core, right increment of the
   support.

In this second case, the parameter increases=TRUE has to be used.

The resampling procedure produces b fuzzy values. During the first step, the four values are randomly generated using the special *w density*: left end of the core, length of the core, left and right
increment of the support. This *w density* is calculated based on the whole fuzzy sample. Then the
new fuzzy variable, which preserves the above-mentioned characteristics, is created. If the parameter b is not specified, it is equal to the length of the initial sample. The output is given in the same
form as the initial sample.

Value

This function returns matrix with b rows of double values. In each row, there is a single resampled
fuzzy number. These fuzzy numbers have the same form as the values from the initial sample
depending on the provided parameter increases.

References

Romaniuk, M., Hryniewicz, O. (2019) Interval-based, nonparametric approach for resampling of
fuzzy numbers Soft Computing, 23 (14), pp. 5883-5903

See Also

ClassicalBootstrap, EWMethod for the EW method, VAFMethod for the VAF method, VAAMethod
for the VAA method

Other resampling functions: ClassicalBootstrap(), DMethod(), EWMethod(), VAAMethod(),
VAFMethod(), VAMethod()

Examples

```
# prepare some fuzzy numbers (first type of the initial sample)

fuzzyValues <- matrix(c(0.25,0.5,1,1.25,0.75,1,1.5,2.2,-1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the w method
```

```
set.seed(12345)

WMethod(fuzzyValues)

WMethod(fuzzyValues,b=4)

# prepare some fuzzy numbers (second type of the initial sample)

fuzzyValuesInc <- matrix(c(0.25,0.5,1,0.25,0.25,1,1.5,0.7,1,0,0,2),
ncol = 4,byrow = TRUE)

# generate the secondary sample using the w method

WMethod(fuzzyValuesInc,increases = TRUE)

WMethod(fuzzyValuesInc,b=4,increases = TRUE)
```

Index