Package 'FuzzyNumbers'

August 29, 2013

Title Tools to deal with fuzzy numbers

Type Package

Description The FuzzyNumbers package provides S4 classes and methods to deal with Fuzzy Numbers: computations of arithmetic operations, approximation by trapezoidal and piecewise linear FNs, visualization, etc.

Version 0.3-1

Date 2013-06-23

URL http://www.rexamine.com/resources/fuzzynumbers/

BugReports https://github.com/Rexamine/FuzzyNumbers/issues

License LGPL (>= 3)

ByteCompile TRUE

Depends R (>= 2.12.0), base, methods, grDevices, graphics, stats

Suggests testthat

Collate 'FuzzyNumbers-package.R' 'class-FuzzyNumber.R''class-DiscontinuousFuzzyNumber.R''class-TrapezoidalFuzzyNumber.R''class-PiecewiseLinearFuzzyNumber.R' 'class-PowerFuzzyNumber.R''methods-core.R' 'methods-supp.R' 'methods-value.R''methods-expectedValue.R' 'methods-weightedExpectedValue.R''methods-width.R' 'methods-ambiguity.R' 'methods-plot.R''methods-as.character.R' 'methods-show.R' 'methods-fapply.R''methods-alphacut.R' 'methods-evaluate.R' 'auxiliary-convert.R''auxiliary-invert.R' 'methods-piecewiseLinearApproximation.R''methods-trapezoidalApproximation.R' 'methods-Arithmetic.R''methods-alphaInterval.R' 'methods-expectedInterval.R''methods-distance.R' 'methods-integrate.R' 'methods-Extract.R''methods-as.PiecewiseLinearFuzzyNumber.R''methods-as.FuzzyNumber.R''methods-as.TrapezoidalFuzzyNumber.R''methods-as.PowerFuzzyNumber.R'

Author Marek Gagolewski [aut, cre], Jan Caha [ctb]

Maintainer Marek Gagolewski <gagolews@rexamine.com>

NeedsCompilation no

Repository CRAN

Date/Publication 2013-06-23 16:16:39

R topics documented:

`FuzzyNumbers-package` *Tools to Deal with Fuzzy Numbers*

Description

FuzzyNumbers is an open source (LGPL 3) package for R. It provides S4 classes and methods to deal with Fuzzy Numbers. The package may be used by the practitioners as well as by the researchers in fuzzy numbers theory (e.g. for testing new algorithms, generating numerical examples, preparing figures).

Details

Fuzzy set theory lets us quite intuitively represent imprecise or vague information. Fuzzy numbers, which form a particular subclass of fuzzy sets of the real line, play a significant role in many important theoretical and/or practical considerations. This is because we often describe our knowledge about objects through numbers, e.g. "I'm about 180 cm tall" or "The rocket was launched between 2 and 3 p.m.".

For the formal definition of a fuzzy number please refer to the `FuzzyNumber` man page. Note that this package also deals with particular types of fuzzy numbers like trapezoidal, piecewise linear, or "parametric" FNs (see `TrapezoidalFuzzyNumber` `PiecewiseLinearFuzzyNumber`, `PowerFuzzyNumber`, and *EXPERIMENTAL* `DiscontinuousFuzzyNumber`)

The package aims to provide the following functionality:

1. Representation of arbitrary fuzzy numbers (including FNs with discontinuous side functions and/or alpha-cuts), as well as their particular types, e.g. trapezoidal and piecewise linear fuzzy numbers,
2. Defuzzification and approximation by triangular and piecewise linear FNs (see e.g. `expectedValue`, `value`, `trapezoidalApproximation`, `piecewiseLinearApproximation`),
3. Visualization of FNs (see `plot`, `as.character`),
4. Basic operations on FNs (see e.g. `fapply`),
5. Aggregation of FNs **TO DO**,
6. Ranking of FNs **TO DO**,
7. Random FN generation **TO DO**,
8. ...

Please feel free to send any comments and feature requests to the author (see his homepage at `http://staff.rexamine.com/gagolews/`).

For a complete list of classes and methods call `help(package="FuzzyNumbers")`. Moreover, you will surely be interested in a step-by-step guide to the package usage and features which is available by calling `vignette('FuzzyNumbers-Tutorial', 'FuzzyNumbers')` or at `http://github.com/Rexamine/FuzzyNumbers/raw/master/inst/doc/FuzzyNumbers-Tutorial.pdf`.

Keywords: Fuzzy Numbers, Fuzzy Sets, Shadowed Sets, Trapezoidal Approximation, Piecewise Linear Approximation, Approximate Reasoning, Imprecision, Vagueness, Randomness.

Acknowledgments: Many thanks to Jan Caha, Przemyslaw Grzegorzewski, Lucian Coroianu, and Pablo Villacorta Iglesias for stimulating discussion.

The development of the package in March-June 2013 was partially supported by the European Union from resources of the European Social Fund, Project PO KL "Information technologies: Research and their interdisciplinary applications", agreement UDA-POKL.04.01.01-00-051/10-00.

Author(s)

Marek Gagolewski <gagolews@rexamine.com>,
with contributions from Jan Caha

References

FuzzyNumbers Homepage, http://www.rexamine.com/resources/fuzzynumbers/.
Ban A.I. (2008), Approximation of fuzzy numbers by trapezoidal fuzzy numbers preserving the expected interval, Fuzzy Sets and Systems 159, pp. 1327-1344.
Ban A.I. (2009), On the nearest parametric approximation of a fuzzy number - Revisited, Fuzzy Sets and Systems 160, pp. 3027–3047.
Chanas S. (2001), On the interval approximation of a fuzzy number, Fuzzy Sets and Systems 122, pp. 353-356.
Coroianu L., Gagolewski M., Grzegorzewski P. (2013), Nearest Piecewise Linear Approximation of Fuzzy Numbers, Fuzzy Sets and Systems, 2013, doi:10.1016/j.fss.2013.02.005.
Delgado M., Vila M.A., Voxman W. (1998), On a canonical representation of a fuzzy number, Fuzzy Sets and Systems 93, pp. 125-135.
Dubois D., Prade H. (1978), Operations on fuzzy numbers, Int. J. Syst. Sci. 9, pp. 613-626.
Dubois D., Prade H. (1987), The mean value of a fuzzy number, Fuzzy Sets and Systems 24, pp. 279-300.
Gagolewski M. (2013), A Guide to the **FuzzyNumbers** Package for R, http://github.com/Rexamine/FuzzyNumbers/raw/master/inst/doc/FuzzyNumbers-Tutorial.pdf.
Grzegorzewski P. (2010), Algorithms for trapezoidal approximations of fuzzy numbers preserving the expected interval, In: Bouchon-Meunier B. et al (Eds.), Foundations of Reasoning Under Uncertainty, Springer, pp. 85-98.
Grzegorzewski P. (1998), Metrics and orders in space of fuzzy numbers, Fuzzy Sets and Systems 97, pp. 83-94.
Grzegorzewski P,. Pasternak-Winiarska K. (2011), Trapezoidal approximations of fuzzy numbers with restrictions on the support and core, Proc. EUSFLAT/LFA 2011, Atlantic Press, pp. 749-756.
Klir G.J., Yuan B. (1995), Fuzzy sets and fuzzy logic. Theory and applications, Prentice Hall, New Jersey.
Stefanini L., Sorini L. (2009), Fuzzy arithmetic with parametric LR fuzzy numbers, In: Proc. IFSA/EUSFLAT 2009, pp. 600-605.
Yeh C.-T. (2008), Trapezoidal and triangular approximations preserving the expected interval, Fuzzy Sets and Systems 159, pp. 1345-1353.

alphacut *Calculate Alpha-Cuts*

Description

If A is a fuzzy numbers, then its α-cuts are always in form of intervals. Moreover, the α-cuts form a nonincreasing chain w.r.t. $alpha$.

Usage

```
  ## S4 method for signature 'FuzzyNumber,numeric'
alphacut(object, alpha)
```

Arguments

object	a fuzzy numbers
alpha	numeric vector with elements in [0,1]

Value

a matrix with two columns (left and right alha cut bounds). if some elements in alpha are not in [0,1], then NA is set.

See Also

Other alpha_cuts: core, supp

Other FuzzyNumber-method: alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Examples

```
A <- TrapezoidalFuzzyNumber(1, 2, 3, 4)
alphacut(A, c(-1, 0.4, 0.2))
```

alphaInterval *Calculate the Alpha-Interval of a Fuzzy Number*

Description

We have $\alpha - Int(A) := [\int_0^1 \alpha A_L(\alpha)\, d\alpha, \int_0^1 \alpha A_U(\alpha)\, d\alpha]$.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
alphaInterval(object, ...)

  ## S4 method for signature 'TrapezoidalFuzzyNumber'
alphaInterval(object)

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber'
alphaInterval(object)

  ## S4 method for signature 'PowerFuzzyNumber'
alphaInterval(object)
```

Arguments

object	a fuzzy number
...	for FuzzyNumber and DiscontinuousFuzzyNumber - additional arguments passed to integrateAlpha

Details

Note that if an instance of the FuzzyNumber or DiscontinuousFuzzyNumber class is given, the calculation is performed via numerical integration. Otherwise, the computation is exact.

Value

numeric vector of length 2

See Also

Other FuzzyNumber-method: alphacut, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: Arithmetic, as.character, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNumber-class, plot

Other PowerFuzzyNumber-method: as.character, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, PowerFuzzyNumber, PowerFuzzyNumber-class

Other TrapezoidalFuzzyNumber-method: Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

---

ambiguity *Calculate the Ambiguity of a Fuzzy Number*

Description

The ambiguity (Delgado et al, 1998) is a measure of nonspecificity of a fuzzy number.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
ambiguity(object, ...)
```

Arguments

object	a fuzzy number
...	additional arguments passed to alphaInterval

Details

The ambiguity is defined as $amb(A) := \int_0^1 \alpha \left(A_U(\alpha) - A_L(\alpha)\right)\, d\alpha$.

Value

a single numeric value

References

Delgado M., Vila M.A., Voxman W. (1998), On a canonical representation of a fuzzy number, Fuzzy Sets and Systems 93, pp. 125-135.

See Also

Other characteristics: expectedValue, value, weightedExpectedValue, width

Other FuzzyNumber-method: alphacut, alphaInterval, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

approxInvert *Approximate the Inverse of a Given Function*

Description

The function may be used to create side generating functions from alpha-cut generators and inversely.

Usage

```
approxInvert(f,
  method = c("monoH.FC", "linear", "hyman"), n = 500)
```

Arguments

f	a monotonic, continuous function f: [0,1]->[0,1]
method	interpolation method: "monoH.FC', "hyman" or "linear"
n	number of interpolation points

Details

The function is a wrapper to splinefun and approxfun. Thus, interpolation is used.

Value

a new function, the approximate inverse of the input

See Also

FuzzyNumber

Other auxiliary: convertAlpha, convertSide

Arithmetic *Arithmetic Operations on Fuzzy Numbers*

Description

Applies arithmetic operations using the extension principle and interval-based calculations.

Usage

```
  ## S4 method for signature 'numeric,FuzzyNumber'
e1 + e2 # e2 + e1

  ## S4 method for signature 'TrapezoidalFuzzyNumber,TrapezoidalFuzzyNumber'
e1 +
    e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,PiecewiseLinearFuzzyNumber'
e1 +
    e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,numeric'
e1 + e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,FuzzyNumber'
e1 +
    e2 # calls as.PiecewiseLinearFuzzyNumber()

  ## S4 method for signature 'numeric,FuzzyNumber'
e1 - e2 # e2*(-1) + e1

  ## S4 method for signature 'TrapezoidalFuzzyNumber,TrapezoidalFuzzyNumber'
e1 -
    e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,PiecewiseLinearFuzzyNumber'
e1 -
    e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,numeric'
e1 - e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,FuzzyNumber'
e1 -
    e2 # calls as.PiecewiseLinearFuzzyNumber()

  ## S4 method for signature 'FuzzyNumber,ANY'
e1 - e2 # -e1

  ## S4 method for signature 'numeric,FuzzyNumber'
e1 * e2 # e2 * e1

  ## S4 method for signature 'TrapezoidalFuzzyNumber,numeric'
e1 * e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,PiecewiseLinearFuzzyNumber'
e1 *
```

```
    e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,FuzzyNumber'
e1 *
    e2 # calls as.PiecewiseLinearFuzzyNumber()

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,numeric'
e1 * e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,numeric'
e1 / e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,PiecewiseLinearFuzzyNumber'
e1 /
    e2

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,FuzzyNumber'
e1 /
    e2 # calls as.PiecewiseLinearFuzzyNumber()
```

Arguments

e1	a fuzzy number or single numeric value
e2	a fuzzy number or single numeric value

Details

Implemented operators: +, -, *, / for piecewise linear fuzzy numbers. Also some versions may be applied on numeric values and trapezoidal fuzzy numbers.

Note that according to the theory the class of PLFNs is not closed under the operations * and /. However, if you operate on a large number of knots, the results should be satisfactory.

Value

a fuzzy number of the class PiecewiseLinearFuzzyNumber of a TrapezoidalFuzzyNumber

a fuzzy number

See Also

Other extension_principle: fapply

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, as.character, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNumber-class, plot

Other TrapezoidalFuzzyNumber-method: alphaInterval, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

---

as.character *Get Basic Information on a Fuzzy Number in a String*

Description

This method is especially useful if you would like to generate LaTeX equations defining a fuzzy numbers.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
as.character(x, toLaTeX=FALSE,
    varnameLaTeX="A")

  ## S4 method for signature 'TrapezoidalFuzzyNumber'
as.character(x,
    toLaTeX=FALSE, varnameLaTeX="A")

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber'
as.character(x,
    toLaTeX=FALSE, varnameLaTeX="A")

  ## S4 method for signature 'PowerFuzzyNumber'
as.character(x,
    toLaTeX=FALSE, varnameLaTeX="A")
```

Arguments

x	a fuzzy number
toLaTeX	logical; should LaTeX code be output?
varnameLaTeX	character; variable name to be included in equations

Details

Consider calling the cat function on the resulting string.

Thanks to Jan Caha for suggesting the toLaTeX arg.

Value

character vector

See Also

Other conversion: as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNumber-class, plot

Other PowerFuzzyNumber-method: alphaInterval, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, PowerFuzzyNumber, PowerFuzzyNumber-class

---

as.FuzzyNumber *Converts an Object to a Fuzzy Number*

Description

Please note that applying this function on a FuzzyNumber child class causes information loss, as it drops all additional slots defined in the child classes. FuzzyNumber is the base class for all FNs. Note that some functions for TFNs or PLFNs work much faster and are more precise. This function shouldn't be used in normal computations.

Usage

```
  ## S4 method for signature 'numeric'
as.FuzzyNumber(object)

  ## S4 method for signature 'FuzzyNumber'
as.FuzzyNumber(object)
```

Arguments

object	a fuzzy number or a single numeric value (crisp number) or vector of length two (interval)

Value

Object of class FuzzyNumber

See Also

Other conversion: as.character, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

---

as.PiecewiseLinearFuzzyNumber

Converts an Object to a Piecewise Linear Fuzzy Number

Description

This method is only for exact conversion. For other cases (e.g. general FNs), use piecewiseLinearApproximation.

Usage

```
  ## S4 method for signature 'TrapezoidalFuzzyNumber'
as.PiecewiseLinearFuzzyNumber(object,
    knot.n=0, knot.alpha=seq(0, 1,
    length.out=knot.n+2)[-c(1,knot.n+2)])

  ## S4 method for signature 'numeric'
as.PiecewiseLinearFuzzyNumber(object,
    knot.n=0, knot.alpha=seq(0, 1,
    length.out=knot.n+2)[-c(1,knot.n+2)])

  ## S4 method for signature 'FuzzyNumber'
as.PiecewiseLinearFuzzyNumber(object,
    knot.n=0, knot.alpha=seq(0, 1,
    length.out=knot.n+2)[-c(1,knot.n+2)])

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber'
as.PiecewiseLinearFuzzyNumber(object,
    knot.n=0, knot.alpha=seq(0, 1,
    length.out=knot.n+2)[-c(1,knot.n+2)])
```

Arguments

object	a fuzzy number or a single numeric value (crisp number) or vector of length two (interval)
knot.n	the number of knots
knot.alpha	knot.n alpha-cut values at knots, defaults to uniformly distributed knots

Value

Object of class PiecewiseLinearFuzzyNumber

See Also

Other conversion: as.character, as.FuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproxima plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNum plot

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

---

as.PowerFuzzyNumber *Converts an Object to a Power Fuzzy Number*

Description

This method is only for exact conversion.

Usage

```
  ## S4 method for signature 'numeric'
as.PowerFuzzyNumber(object)

  ## S4 method for signature 'FuzzyNumber'
as.PowerFuzzyNumber(object)

  ## S4 method for signature 'PowerFuzzyNumber'
as.PowerFuzzyNumber(object)

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber'
as.PowerFuzzyNumber(object)

  ## S4 method for signature 'TrapezoidalFuzzyNumber'
as.PowerFuzzyNumber(object)
```

Arguments

object	a fuzzy number or a single numeric value (crisp number) or vector of length two (interval)

Value

Object of class PowerFuzzyNumber

See Also

Other conversion: as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.TrapezoidalFuzzyNumber

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PiecewiseLinearFuzzyN as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNum plot

Other PowerFuzzyNumber-method: alphaInterval, as.character, as.TrapezoidalFuzzyNumber, expectedInterval, PowerFuzzyNumber, PowerFuzzyNumber-class

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

---

as.TrapezoidalFuzzyNumber

Converts an Object to a Trapezoidal Fuzzy Number

Description

This method is only for exact conversion. For other cases (e.g. general FNs), use trapezoidalApproximation.

Usage

```
  ## S4 method for signature 'numeric'
as.TrapezoidalFuzzyNumber(object)

  ## S4 method for signature 'FuzzyNumber'
as.TrapezoidalFuzzyNumber(object)

  ## S4 method for signature 'PowerFuzzyNumber'
as.TrapezoidalFuzzyNumber(object)

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber'
as.TrapezoidalFuzzyNumber(object)

  ## S4 method for signature 'TrapezoidalFuzzyNumber'
as.TrapezoidalFuzzyNumber(object)
```

Arguments

object	a fuzzy number or a single numeric value (crisp number) or vector of length two (interval)

Value

Object of class TrapezoidalFuzzyNumber

See Also

Other conversion: as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PiecewiseLinearFuzzyN as.PowerFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNumber-cl plot

Other PowerFuzzyNumber-method: alphaInterval, as.character, as.PowerFuzzyNumber, expectedInterval, PowerFuzzyNumber, PowerFuzzyNumber-class

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

---

convertAlpha *Convert a Given Upper/Lower Alpha-Cut Function to an Alpha-Cut Generating Function*

Description

The resulting function calls the original function and then linearly scales its output.

Usage

```
convertAlpha(f, y1, y2)
```

Arguments

f	a function into [y1,y2]
y1	numeric vector of length 1
y2	numeric vector of length 1

Value

a new function defined on [0,1] (scaled input)

See Also

FuzzyNumber

Other auxiliary: approxInvert, convertSide

---

convertSide *Convert a Given Side Function to Side Generating Function*

Description

The resulting function linearly scales the input and passes it to the original function.

Usage

```
convertSide(f, x1, x2)
```

Arguments

f	a function defined on [x1,x2]
x1	numeric vector of length 1; if longer, only the first element is used
x2	numeric vector of length 1; if longer, only the first element is used

Details

The function works for x1<x2 and x1>x2.

Value

a new function defined on [0,1] (scaled input)

See Also

FuzzyNumber

Other auxiliary: approxInvert, convertAlpha

core *Calculate the Core of a Fuzzy Number*

Description

We have $\mathrm{core}(A) := [a2, a3]$. This gives the values that a fuzzy number necessarily represents.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
core(object)
```

Arguments

object	a fuzzy number

Value

numeric vector of length 2

See Also

Other alpha_cuts: alphacut, supp

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

DiscontinuousFuzzyNumber *Creates a Fuzzy Number with Possibly Discontinuous Side Functions or Alpha-Cut Bounds*

Description

For convenience, objects of class DiscontinuousFuzzyNumber may be created with this function.

Usage

```
DiscontinuousFuzzyNumber(a1, a2, a3, a4,
  lower = function(a) rep(NA_real_, length(a)),
  upper = function(a) rep(NA_real_, length(a)),
  left = function(x) rep(NA_real_, length(x)),
  right = function(x) rep(NA_real_, length(x)),
  discontinuities.left = numeric(0),
  discontinuities.right = numeric(0),
  discontinuities.lower = numeric(0),
  discontinuities.upper = numeric(0))
```

Arguments

a1	a number specyfing left bound of the support
a2	a number specyfing left bound of the core
a3	a number specyfing right bound of the core
a4	a number specyfing right bound of the support
lower	lower alpha-cut bound generator; a nondecreasing function [0,1]->[0,1] or returning NA_real_
upper	upper alpha-cut bound generator; a nonincreasing function [0,1]->[1,0] or returning NA_real_
left	lower side function generator; a nondecreasing function [0,1]->[0,1] or returning NA_real_
right	upper side function generator; a nonincreasing function [0,1]->[1,0] or returning NA_real_
discontinuities.left	nondecreasingly sorted numeric vector with elements in (0,1), possibly of length 0
discontinuities.right	nondecreasingly sorted numeric vector with elements in (0,1), possibly of length 0
discontinuities.lower	nondecreasingly sorted numeric vector with elements in (0,1), possibly of length 0
discontinuities.upper	nondecreasingly sorted numeric vector with elements in (0,1), possibly of length 0

Value

Object of class DiscontinuousFuzzyNumber

See Also

Other DiscontinuousFuzzyNumber-method: DiscontinuousFuzzyNumber-class, distance, integrateAlpha, plot

DiscontinuousFuzzyNumber-class ***EXPERIMENTAL** S4 Class Representing a Fuzzy Number with Discontinuous Side Functions or Alpha-Cut Bounds*

Description

Discontinuity information increase the precision of some numerical integration-based algorithms, e.g. of `piecewiseLinearApproximation`. It also allows for making more valid fuzzy number plots.

Slots

`a1`, `a2`, `a3`, `a4`, `lower`, `upper`, `left`, `right`: Inherited from the `FuzzyNumber` class.

`discontinuities.left`: nondecreasingly sorted numeric vector with elements in (0,1); discontinuity points for the left side generating function

`discontinuities.right`: nondecreasingly sorted numeric vector with elements in (0,1); discontinuity points for the right side generating function

`discontinuities.lower`: nondecreasingly sorted numeric vector with elements in (0,1); discontinuity points for the lower alpha-cut bound generator

`discontinuities.upper`: nondecreasingly sorted numeric vector with elements in (0,1); discontinuity points for the upper alpha-cut bound generator

Extends

Class `FuzzyNumber`, directly.

See Also

`DiscontinuousFuzzyNumber` for a convenient constructor

Other DiscontinuousFuzzyNumber-method: `DiscontinuousFuzzyNumber`, `distance`, `integrateAlpha`, `plot`

Examples

```
showClass("DiscontinuousFuzzyNumber")
showMethods(classes="DiscontinuousFuzzyNumber")
```

distance *Calculate the Distance Between Two Fuzzy Numbers*

Description

Currently, only Euclidean distance may be calculated. We have $d_E^2(A,B) := \int_0^1 (A_L(\alpha) - B_L(\alpha))^2 \, d\alpha, \int_0^1 + (A_U(\alpha) - B_U(\alpha))^2 \, d\alpha$, see (Grzegorzewski, 1988).

Usage

```
  ## S4 method for signature 'FuzzyNumber,FuzzyNumber'
distance(e1, e2,
    type=c("Euclidean", "EuclideanSquared"), ...)

  ## S4 method for signature 'FuzzyNumber,DiscontinuousFuzzyNumber'
distance(e1,
    e2, type=c("Euclidean", "EuclideanSquared"), ...)

  ## S4 method for signature 'DiscontinuousFuzzyNumber,FuzzyNumber'
distance(e1,
    e2, type=c("Euclidean", "EuclideanSquared"), ...)

  ## S4 method for signature 'DiscontinuousFuzzyNumber,DiscontinuousFuzzyNumber'
distance(e1,
    e2, type=c("Euclidean", "EuclideanSquared"), ...)
```

Arguments

e1	a fuzzy number
e2	a fuzzy number
type	one of "Euclidean", "EuclideanSquared"
...	additional arguments passed to integrate

Details

The calculation are done using numerical integration,

Value

the calculated distance, single numeric value

References

Grzegorzewski P., Metrics and orders in space of fuzzy numbers, Fuzzy Sets and Systems 97, 1998, pp. 83-94.

See Also

Other DiscontinuousFuzzyNumber-method: DiscontinuousFuzzyNumber, DiscontinuousFuzzyNumber-class, integrateAlpha, plot

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

---

evaluate *Evaluate the Membership Function*

Description

This function returns the value(s) of the membership function of a fuzzy number at given point(s).

Usage

```
  ## S4 method for signature 'FuzzyNumber,numeric'
evaluate(object, x)
```

Arguments

object	a fuzzy numbers
x	numeric vector

Value

a numeric vector

See Also

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Examples

```
T <- TrapezoidalFuzzyNumber(1,2,3,4)
evaluate(T, seq(0, 5, by=0.5))
```

`expectedInterval` *Calculate the Expected Interval of a Fuzzy Number*

Description

We have $EI(A) := [\int_0^1 A_L(\alpha)\, d\alpha, \int_0^1 A_U(\alpha)\, d\alpha]$, see (Duboid, Prade, 1987).

Usage

```
  ## S4 method for signature 'FuzzyNumber'
expectedInterval(object, ...)

  ## S4 method for signature 'TrapezoidalFuzzyNumber'
expectedInterval(object)

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber'
expectedInterval(object)

  ## S4 method for signature 'PowerFuzzyNumber'
expectedInterval(object)
```

Arguments

`object`	a fuzzy number
`...`	for `FuzzyNumber` and `DiscontinuousFuzzyNumber` - additional arguments passed to `integrateAlpha`

Details

Note that if an instance of the `FuzzyNumber` or `DiscontinuousFuzzyNumber` class is given, the calculation is performed via numerical integration. Otherwise, the computation is exact.

Value

numeric vector of length 2

References

Dubois D., Prade H. (1987), The mean value of a fuzzy number, Fuzzy Sets and Systems 24, pp. 279-300.

See Also

Other FuzzyNumber-method: `alphacut`, `alphaInterval`, `ambiguity`, `Arithmetic`, `as.character`, `as.FuzzyNumber`, `as.PiecewiseLinearFuzzyNumber`, `as.PowerFuzzyNumber`, `as.TrapezoidalFuzzyNumber`, `core`, `distance`, `evaluate`, `expectedValue`, `FuzzyNumber`, `FuzzyNumber-class`, `integrateAlpha`,

piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PiecewiseLinearFuzzyN as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNumber-class, plot

Other PowerFuzzyNumber-method: alphaInterval, as.character, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumb PowerFuzzyNumber, PowerFuzzyNumber-class

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-cla TriangularFuzzyNumber

---

`expectedValue` *Calculate the Expected Value of a Fuzzy Number*

Description

The calculation of the so-called expected value is one of possible methods to deffuzify a fuzzy number.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
expectedValue(object, ...)
```

Arguments

`object`	a fuzzy number
`...`	additional arguments passed to `expectedInterval`

Details

The expected value of A is defined as $EV(A) := (EI_U(A) + EI_L(A))/2$, where EI is the `expectedInterval`.

Value

a single numeric value

See Also

Other characteristics: ambiguity, value, weightedExpectedValue, width

Other deffuzification: value, weightedExpectedValue

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Extract *FuzzyNumber Slot Accessors*

Description

For possible slot names see man pages for the FuzzyNumber class and its derivatives

Usage

```
  ## S4 method for signature 'FuzzyNumber,character'
x[i]

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,character'
x[i]

  ## S4 method for signature 'PowerFuzzyNumber,character'
x[i]

  ## S4 method for signature 'DiscontinuousFuzzyNumber,character'
x[i]
```

Arguments

x	a fuzzy number
i	character; slot name

Details

All slot accessors are read-only.

Value

slot value

Examples

```
A <- FuzzyNumber(1,2,3,4)
A["a1"]
A["right"]
```

`fapply` *Apply a Function on a Fuzzy Number*

Description

Applies a given monotonic function using the extension principle (i.e. the function is applied on alpha-cuts).

Usage

```
  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,function'
fapply(object,
    fun, ...)
```

Arguments

`object`	a fuzzy number
`fun`	a monotonic, vectorized R function
`...`	additional arguments passed to `fun`

Details

Currently only a method for the PiecewiseLinearFuzzyNumber class has been defined. The computations are exact (up to a numeric error) at knots. So, make sure you have a sufficient number of knots if you want good approximation.

For other types of fuzzy numbers, consider using `piecewiseLinearApproximation`.

Value

a PiecewiseLinearFuzzyNumber

See Also

Other extension_principle: `Arithmetic`

Other PiecewiseLinearFuzzyNumber-method: `alphaInterval`, `Arithmetic`, `as.character`, `as.PiecewiseLinearFuzzyN`
`as.PowerFuzzyNumber`, `as.TrapezoidalFuzzyNumber`, `expectedInterval`, `PiecewiseLinearFuzzyNumber`,
`PiecewiseLinearFuzzyNumber-class`, `plot`

FuzzyNumber *Creates a Fuzzy Number*

Description

For convenience, objects of class FuzzyNumber may be created with this function.

Usage

```
FuzzyNumber(a1, a2, a3, a4,
  lower = function(a) rep(NA_real_, length(a)),
  upper = function(a) rep(NA_real_, length(a)),
  left = function(x) rep(NA_real_, length(x)),
  right = function(x) rep(NA_real_, length(x)))
```

Arguments

a1	a number specyfing left bound of the support
a2	a number specyfing left bound of the core
a3	a number specyfing right bound of the core
a4	a number specyfing right bound of the support
lower	lower alpha-cut bound generator; a nondecreasing function [0,1]->[0,1] or returning NA_real_
upper	upper alpha-cut bound generator; a nonincreasing function [0,1]->[1,0] or returning NA_real_
left	lower side function generator; a nondecreasing function [0,1]->[0,1] or returning NA_real_
right	upper side function generator; a nonincreasing function [0,1]->[1,0] or returning NA_real_

Value

Object of class FuzzyNumber

See Also

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

`FuzzyNumber-class` *S4 class Representing a Fuzzy Number*

Description

Formally, a fuzzy number A (Dubois, Prade, 1978) is a fuzzy subset of the real line R with membership function μ given by:

$$\mu(x) = \begin{array}{ll} \mid 0 & \text{if } x < a1, \\ \mid left((x-a1)/(a2-a1)) & \text{if } a1 \le x < a2, \\ \mid 1 & \text{if } a2 \le x \le a3, \\ \mid right((x-a3)/(a4-a3)) & \text{if } a3 < x \le a4, \\ \mid 0 & \text{if } a4 < x, \end{array}$$

where $a1, a2, a3, a4 \in R$, $a1 \le a2 \le a3 \le a4$, $left : [0,1] \to [0,1]$ is a nondecreasing function called the *left side generator of* A, and $right : [0,1] \to [0,1]$ is a nonincreasing function called the *right side generator of* A.

Alternatively, it may be shown that each fuzzy number A may be uniquely determined by specifying its α-cuts, $A(\alpha)$. We have $A(0) = [a1, a4]$ and

$$A(\alpha) = [a1 + (a2 - a1) * lower(\alpha), a3 + (a4 - a3) * upper(\alpha)]$$

for $0 < \alpha \le 1$, where $lower : [0,1] \to [0,1]$ and $upper : [0,1] \to [0,1]$ are, respectively, strictly increasing and decreasing functions satisfying $lower(\alpha) = \inf\{x : \mu(x) \ge \alpha\}$ and $upper(\alpha) = \sup\{x : \mu(x) \ge \alpha\}$.

Details

Please note that many algorithms that deal with fuzzy numbers often use α-cuts rather than side functions.

Note that the **FuzzyNumbers** package also deals with particular types of fuzzy numbers like trapezoidal, piecewise linear, or "parametric" FNs.

Slots

`a1`: Single numeric value specifying the left bound for the support.

`a2`: Single numeric value specifying the left bound for the core.

`a3`: Single numeric value specifying the right bound for the core.

`a4`: Single numeric value specifying the right bound for the support.

`lower`: A nondecreasing function [0,1]->[0,1] that gives the lower alpha-cut bound.

`upper`: A nonincreasing function [0,1]->[1,0] that gives the upper alpha-cut bound.

`left`: A nondecreasing function [0,1]->[0,1] that gives the left side function.

`right`: A nonincreasing function [0,1]->[1,0] that gives the right side function.

Child/sub classes

TrapezoidalFuzzyNumber, PiecewiseLinearFuzzyNumber, PowerFuzzyNumber, and DiscontinuousFuzzyNumber

See Also

FuzzyNumber for a convenient constructor, and as.FuzzyNumber for conversion of objects to this class. Also, see convertSide for creating side functions generators, convertAlpha for creating alpha-cut bounds generators, approxInvert for inverting side functions/alpha-cuts numerically.

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Examples

```
showClass("FuzzyNumber")
showMethods(classes="FuzzyNumber")
```

---

integrateAlpha *Numerically Integrate Alpha-Cut Bounds*

Description

Integrates numerically a transformed or weighted lower or upper alpha-cut bound of a fuzzy number.

Usage

```
  ## S4 method for signature 'FuzzyNumber,character,numeric,numeric'
integrateAlpha(object,
    which=c("lower", "upper"), from=0, to=1, weight=NULL,
    transform=NULL, ...)

  ## S4 method for signature 'DiscontinuousFuzzyNumber,character,numeric,numeric'
integrateAlpha(object,
    which=c("lower", "upper"), from=0, to=1, weight=NULL,
    transform=NULL, ...)
```

Arguments

object	a fuzzy number
which	one of "lower", "upper"
from	numeric
to	numeric
weight	a function or NULL
transform	a function or NULL
...	additional arguments passed to integrate or integrate_discont_val

Value

a single numeric value

See Also

Other DiscontinuousFuzzyNumber-method: DiscontinuousFuzzyNumber, DiscontinuousFuzzyNumber-class, distance, plot

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

---

integrate_discont_val *Integrate a Function with at Most Finite Number of Discontinuities *EXPERIMENTAL**

Description

The function uses multiple calls to integrate.

Usage

```
integrate_discont_val(f, from, to,
  discontinuities = numeric(0), ...)
```

Arguments

f	an R function taking a numeric vector of length 1 as its first argument and returning a numeric vector of length 1
from	the lower limit of integration
to	the upper limit of integration
discontinuities	nondecreasingly sorted numeric vector which indicates the points at which f is discontinuous
...	further arguments to be passed to the integrate function.

Value

the estimate of the integral

`piecewiseLinearApproximation` *Piecewise Linear Approximation of a Fuzzy Number*

Description

This method finds a piecewise linear approximation $P(A)$ of a given fuzzy number A by using the algorithm specified by the `method` parameter.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
piecewiseLinearApproximation(object,
    method=c("NearestEuclidean", "Naive",
    "ApproximateNearestEuclidean [DEPRECATED]"), knot.n=1,
    knot.alpha=seq(0, 1,
    length.out=knot.n+2)[-c(1,knot.n+2)], ...,
    verbose=FALSE)
```

Arguments

`object`	a fuzzy number
`method`	character; one of: `"NearestEuclidean"` (default), or `"Naive"`
`knot.n`	desired number of knots (if missing, then calculated from given `knot.alpha`)
`knot.alpha`	alpha-cuts at which knots will be positioned (defaults to equally distributed knots)
`verbose`	logical; should some technical details on the computations being performed be printed? [only `"NearestEuclidean"`]
`...`	further arguments passed to `integrateAlpha` [only `"NearestEuclidean"`]

Details

'`method`' may be one of:

1. `NearestEuclidean`: see (Coroianu, Gagolewski, Grzegorzewski, 2013a) for the description of the `knot.n==1` case; uses numerical integration, see `integrateAlpha`. Slow for large `knot.n`.
2. `Naive`: We have core(A)==core(T(A)) and supp(A)==supp(T(A)) and the knots are taken directly from the specified alpha cuts (linear interpolation).

References

Coroianu L., Gagolewski M., Grzegorzewski P. (2013a), Nearest Piecewise Linear Approximation of Fuzzy Numbers, Fuzzy Sets and Systems, doi:10.1016/j.fss.2013.02.005.
Coroianu L., Gagolewski M., Grzegorzewski P. (2013b), Nearest Piecewise Linear Approximation of Fuzzy Numbers - general case, in preparation.

See Also

Other approximation: trapezoidalApproximation

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Examples

```
(A <- FuzzyNumber(-1,0,1,3,lower=function(x) sqrt(x),upper=function(x) 1-sqrt(x)))
(PA <- piecewiseLinearApproximation(A, "NearestEuclidean", knot.n=1, knot.alpha=0.2))
```

PiecewiseLinearFuzzyNumber

Creates a Piecewise Linear Fuzzy Number

Description

For convenience, objects of class PiecewiseLinearFuzzyNumber may be created with this function.

Usage

```
PiecewiseLinearFuzzyNumber(a1, a2, a3, a4, knot.n = 0,
  knot.alpha = numeric(0), knot.left = numeric(0),
  knot.right = numeric(0))
```

Arguments

a1	a number specyfing left bound of the support
a2	a number specyfing left bound of the core
a3	a number specyfing right bound of the core
a4	a number specyfing right bound of the support
knot.n	the number of knots
knot.alpha	knot.n alpha-cut values at knots
knot.left	knot.n knots on the left side; a nondecreasingly sorted vector with elements in [a1,a2]
knot.right	knot.n knots on the right side; a nondecreasingly sorted vector with elements in [a3,a4]

Details

If a1, a2, a3, and a4 are missing, then knot.left and knot.right may be of length knot.n+2.

If knot.n is not given, then it guessed from length(knot.left). If knot.alpha is missing, then the knots will be equally distributed on the interval [0,1].

Value

An object of class PiecewiseLinearFuzzyNumber.

See Also

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PiecewiseLinearFuzzyN
as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber-cla
plot

PiecewiseLinearFuzzyNumber-class

S4 Class Representing a Piecewise Linear Fuzzy Number

Description

A piecewise linear fuzzy number (PLFN) has side functions and alpha-cut bounds that linearly interpolate a given set of points (at fixed alpha-cuts).

Details

If knot.n is equal to 0 or all left and right knots lie on common lines, then a Piecewise Linear Fuzzy Number reduces to a TrapezoidalFuzzyNumber. Note that, however, the TrapezoidalFuzzyNumber does not inherit from PiecewiseLinearFuzzyNumber for efficiency reasons. To convert the former to the latter, call as.PiecewiseLinearFuzzyNumber.

Slots

a1, a2, a3, a4, lower, upper, left, right: Inherited from the FuzzyNumber class.

knot.n: number of knots, a single integer value, 0 for a trapezoidal fuzzy number

knot.alpha: alpha-cuts, increasingly sorted vector of length knot.n with elements in [0,1]

knot.left: nondecreasingly sorted vector of length knot.n; defines left alpha-cut bounds at knots

knot.right: nondecreasingly sorted vector of length knot.n; defines right alpha-cut bounds at knots

Extends

Class FuzzyNumber, directly.

See Also

PiecewiseLinearFuzzyNumber for a convenient constructor, as.PiecewiseLinearFuzzyNumber for conversion of objects to this class, and piecewiseLinearApproximation for approximation routines.

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PiecewiseLinearFuzzyN
as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber,
plot

Examples

```
showClass("PiecewiseLinearFuzzyNumber")
showMethods(classes="PiecewiseLinearFuzzyNumber")
```

---

plot *Plot a Fuzzy Number*

Description

The function aims to provide a similar look-and-feel to the built-in `plot.default` and `curve` function.

Usage

```
  ## S4 method for signature 'FuzzyNumber,missing'
plot(x, y, from=NULL,
    to=NULL, n=101, at.alpha=NULL,
    draw.membership.function=TRUE,
    draw.alphacuts=!draw.membership.function, xlab=NULL,
    ylab=NULL, xlim=NULL, ylim=NULL, type="l", col=1,
    lty=1, pch=1, lwd=1, shadowdensity=15, shadowangle=45,
    shadowcol=col, shadowborder=NULL, add=FALSE, ...)

  ## S4 method for signature 'TrapezoidalFuzzyNumber,missing'
plot(x, y,
    from=NULL, to=NULL, draw.membership.function=TRUE,
    draw.alphacuts=!draw.membership.function, xlab=NULL,
    ylab=NULL, xlim=NULL, ylim=NULL, type="l", col=1,
    lty=1, pch=1, lwd=1, add=FALSE, ...)

  ## S4 method for signature 'PiecewiseLinearFuzzyNumber,missing'
plot(x, y,
    from=NULL, to=NULL, draw.membership.function=TRUE,
    draw.alphacuts=!draw.membership.function, xlab=NULL,
    ylab=NULL, xlim=NULL, ylim=NULL, type="l", col=1,
    lty=1, pch=1, lwd=1, add=FALSE, ...)

  ## S4 method for signature 'DiscontinuousFuzzyNumber,missing'
plot(x, y,
    from=NULL, to=NULL, n=101,
    draw.membership.function=TRUE,
    draw.alphacuts=!draw.membership.function, xlab=NULL,
    ylab=NULL, xlim=NULL, ylim=NULL, type="l", col=1,
    lty=1, pch=1, lwd=1, add=FALSE, ...)
```

Arguments

x	a fuzzy number
y	not used
from	numeric;
to	numeric;
n	numeric; number of points to probe
at.alpha	numeric vector; give exact alpha-cuts at which linear interpolation should be done
draw.membership.function	logical; you want membership function (TRUE) or alpha-cuts plot (FALSE)?
draw.alphacuts	logical; defaults !draw.membership.function
xlab	character; x-axis label
ylab	character; y-axis label
xlim	numeric;
ylim	numeric;
type	character; defaults "l"; plot type, e.g.~"l" for lines, "p" for points, or "b" for both
col	see plot.default
lty	see plot.default
pch	see plot.default
lwd	see plot.default
shadowdensity	numeric; for shadowed sets;
shadowangle	numeric; for shadowed sets;
shadowcol	color specification, see plot.default; for shadowed sets;
shadowborder	numeric; for shadowed sets;
add	logical; add another FuzzyNumber to existing plot?
...	further arguments passed to plot.default

Details

Note that if from > a1 then it is set to a1.

Value

nothing really interesting

See Also

Other DiscontinuousFuzzyNumber-method: DiscontinuousFuzzyNumber, DiscontinuousFuzzyNumber-class, distance, integrateAlpha

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, show, supp, trapezoidalApproximation, value, weightedExpectedValue, width

Other PiecewiseLinearFuzzyNumber-method: alphaInterval, Arithmetic, as.character, as.PiecewiseLinearFuzzyN as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, fapply, PiecewiseLinearFuzzyNumber, PiecewiseLinearFuzzyNumber-class

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

Examples

```
plot(FuzzyNumber(0,1,2,3), col="gray")
plot(FuzzyNumber(0,1,2,3, left=function(x) x^2, right=function(x) 1-x^3), add=TRUE)
plot(FuzzyNumber(0,1,2,3, lower=function(x) x, upper=function(x) 1-x), add=TRUE, col=2)
```

PowerFuzzyNumber *Creates a Fuzzy Number with Sides Given by Power Functions*

Description

For convenience, objects of class PowerFuzzyNumber may be created with this function.

Usage

```
PowerFuzzyNumber(a1, a2, a3, a4, p.left = 1, p.right = 1)
```

Arguments

a1	a number specyfing left bound of the support
a2	a number specyfing left bound of the core
a3	a number specyfing right bound of the core
a4	a number specyfing right bound of the support
p.left	a positive number specyfing the exponent for the left side
p.right	a positive number specyfing the exponent for the right side

Value

Object of class PowerFuzzyNumber

See Also

Other PowerFuzzyNumber-method: alphaInterval, as.character, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumb
expectedInterval, PowerFuzzyNumber-class

PowerFuzzyNumber-class

S4 class Representing a Fuzzy Number with Sides Given by Power Functions

---

Description

Fuzzy numbers which sides are given by power functions are defined using four coefficients a1 <= a2 <= a3 <= a4, and parameters p.left, p.right>0, which determine exponents for the side functions.

Details

We have $\texttt{left}(x) = x^{\texttt{p.left}}$, and $\texttt{right}(x) = (1-x)^{\texttt{p.right}}$.

This class is a natural generalization of trapezoidal FNs. For other see PiecewiseLinearFuzzyNumber.

Slots

a1, a2, a3, a4, lower, upper, left, right: Inherited from the FuzzyNumber class.

p.left: single numeric value; 1.0 for a trapezoidal FN.

p.right: single numeric value; 1.0 for a trapezoidal FN.

Extends

Class FuzzyNumber, directly.

See Also

PowerFuzzyNumber for a convenient constructor, as.PowerFuzzyNumber for conversion of objects to this class.

PowerFuzzyNumber for a convenient constructor

Other PowerFuzzyNumber-method: alphaInterval, as.character, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumb
expectedInterval, PowerFuzzyNumber

Examples

```
showClass("PowerFuzzyNumber")
showMethods(classes="PowerFuzzyNumber")
```

show *Print Basic Information on a Fuzzy Number*

Description

See `as.character` for more details.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
show(object)
```

Arguments

`object`	a fuzzy number

Details

The method `as.character` is called on given fuzzy number object with default arguments. The results are printed on `stdout`.

Value

nothing interesting

See Also

Other FuzzyNumber-method: `alphacut`, `alphaInterval`, `ambiguity`, `Arithmetic`, `as.character`, `as.FuzzyNumber`, `as.PiecewiseLinearFuzzyNumber`, `as.PowerFuzzyNumber`, `as.TrapezoidalFuzzyNumber`, `core`, `distance`, `evaluate`, `expectedInterval`, `expectedValue`, `FuzzyNumber`, `FuzzyNumber-class`, `integrateAlpha`, `piecewiseLinearApproximation`, `plot`, `supp`, `trapezoidalApproximation`, `value`, `weightedExpectedValue`, `width`

supp *Calculate the Support of a Fuzzy Number*

Description

We have $\mathrm{supp}(A) := [a1, a4]$. This gives the values that a fuzzy number possibly may represent.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
supp(object)
```

Arguments

object	a fuzzy number

Value

numeric vector of length 2

See Also

Other alpha_cuts: alphacut, core

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, trapezoidalApproximation, value, weightedExpectedValue, width

---

trapezoidalApproximation

Trapezoidal Approximation of a Fuzzy Number

Description

This method finds a trapezoidal approximation $T(A)$ of a given fuzzy number A by using the algorithm specified by the method parameter.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
trapezoidalApproximation(object,
    method=c("NearestEuclidean",
    "ExpectedIntervalPreserving", "SupportCoreRestricted",
    "Naive"), ..., verbose=FALSE)
```

Arguments

object	a fuzzy number
method	character; one of: "NearestEuclidean" (default), "ExpectedIntervalPreserving", "SupportCoreRestricted", "Naive"
verbose	logical; should some technical details on the computations being performed be printed?
...	further arguments passed to integrateAlpha

Details

`method` may be one of:

1. `NearestEuclidean`: see (Ban, 2009); uses numerical integration, see `integrateAlpha`
2. `Naive`: We have core(A)==core(T(A)) and supp(A)==supp(T(A))
3. `ExpectedIntervalPreserving`: L2-nearest trapezoidal approximation preserving the expected interval given in (Grzegorzewski, 2010; Ban, 2008; Yeh, 2008) Unfortunately, for highly skewed membership functions this approximation operator may have quite unfavourable behavior. E.g. if Val(A) < EV_1/3(A) or Val(A) > EV_2/3(A), then it may happen that the core of the output and the core of the original fuzzy number A are disjoint (cf. Grzegorzewski, Pasternak-Winiarska, 2011)
4. `SupportCoreRestricted`: This method was proposed in (Grzegorzewski, Pasternak-Winiarska, 2011). L2-nearest trapezoidal approximation with constraints core(A) $\subseteq$ core(T(A)) and supp(T(A)) $\subseteq$ supp(A), i.e. for which each point that surely belongs to A also belongs to T(A), and each point that surely does not belong to A also does not belong to T(A).

References

Ban A.I. (2008), Approximation of fuzzy numbers by trapezoidal fuzzy numbers preserving the expected interval, Fuzzy Sets and Systems 159, pp. 1327-1344.
Ban A.I. (2009), On the nearest parametric approximation of a fuzzy number - Revisited, Fuzzy Sets and Systems 160, pp. 3027–3047.
Grzegorzewski P. (2010), Algorithms for trapezoidal approximations of fuzzy numbers preserving the expected interval, In: Bouchon-Meunier B. et al (Eds.), Foundations of Reasoning Under Uncertainty, Springer, pp. 85-98.
Grzegorzewski P, Pasternak-Winiarska K. (2011), Trapezoidal approximations of fuzzy numbers with restrictions on the support and core, Proc. EUSFLAT/LFA 2011, Atlantic Press, pp. 749-756.
Yeh C.-T. (2008), Trapezoidal and triangular approximations preserving the expected interval, Fuzzy Sets and Systems 159, pp. 1345-1353.

See Also

Other approximation: `piecewiseLinearApproximation`

Other FuzzyNumber-method: `alphacut`, `alphaInterval`, `ambiguity`, `Arithmetic`, `as.character`, `as.FuzzyNumber`, `as.PiecewiseLinearFuzzyNumber`, `as.PowerFuzzyNumber`, `as.TrapezoidalFuzzyNumber`, `core`, `distance`, `evaluate`, `expectedInterval`, `expectedValue`, `FuzzyNumber`, `FuzzyNumber-class`, `integrateAlpha`, `piecewiseLinearApproximation`, `plot`, `show`, `supp`, `value`, `weightedExpectedValue`, `width`

Examples

```
(A <- FuzzyNumber(-1,0,1,40,lower=function(x) sqrt(x),upper=function(x) 1-sqrt(x)))
(TA <- trapezoidalApproximation(A, "ExpectedIntervalPreserving")) # Note that cores are disjoint!
expectedInterval(A)
expectedInterval(TA)
```

TrapezoidalFuzzyNumber

Creates a Trapezoidal Fuzzy Number

Description

For convenience, objects of class TrapezoidalFuzzyNumber may be created with this function.

Usage

```
TrapezoidalFuzzyNumber(a1, a2, a3, a4)
```

Arguments

a1	a number specyfing left bound of the support
a2	a number specyfing left bound of the core
a3	a number specyfing right bound of the core
a4	a number specyfing right bound of the support

Value

Object of class TrapezoidalFuzzyNumber

See Also

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber-class, TriangularFuzzyNumber

TrapezoidalFuzzyNumber-class

S4 class Representing a Trapezoidal Fuzzy Number

Description

Trapezoidal Fuzzy Numbers have linear side functions and alpha-cut bounds.

Details

Trapezoidal fuzzy numbers are among the simplest FNs. Despite their simplicity, however, they include triangular FNs, "crisp" real intervals, and "crisp" reals. Please note that currently no separate classes for these particular TFNs types are implemented in the package.

Slots

`a1`, `a2`, `a3`, `a4`, `lower`, `upper`, `left`, `right`: Inherited from the `FuzzyNumber` class.

Extends

Class `FuzzyNumber`, directly.

See Also

`TrapezoidalFuzzyNumber` for a convenient constructor, `as.TrapezoidalFuzzyNumber` for conversion of objects to this class, and `trapezoidalApproximation` for approximation routines.

Other TrapezoidalFuzzyNumber-method: `alphaInterval`, `Arithmetic`, `as.PiecewiseLinearFuzzyNumber`, `as.PowerFuzzyNumber`, `as.TrapezoidalFuzzyNumber`, `expectedInterval`, `plot`, `TrapezoidalFuzzyNumber`, `TriangularFuzzyNumber`

Examples

```
showClass("TrapezoidalFuzzyNumber")
showMethods(classes="TrapezoidalFuzzyNumber")
```

TriangularFuzzyNumber *Creates a Triangular Fuzzy Number*

Description

For convenience, objects of class `TrapezoidalFuzzyNumber` may be created with this function.

Usage

```
TriangularFuzzyNumber(a1, amid, a4)
```

Arguments

`a1`	a number specyfing left bound of the support
`amid`	a number specyfing the core
`a4`	a number specyfing right bound of the support

Details

Currently there is no separate class of a Triangular Fuzzy Number.

Value

Object of class `TrapezoidalFuzzyNumber`

See Also

Other TrapezoidalFuzzyNumber-method: alphaInterval, Arithmetic, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, expectedInterval, plot, TrapezoidalFuzzyNumber, TrapezoidalFuzzyNumber-class

---

value *Calculate the Value of a Fuzzy Number*

Description

The calculation of the so-called value is one of possible methods to deffuzify a fuzzy number.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
value(object, ...)
```

Arguments

object	a fuzzy number
...	additional arguments passed to alphaInterval

Details

The value of A (Delgrado et al, 1998) is defined as $val(A) := \int_0^1 \alpha \left(A_L(\alpha) + A_U(\alpha)\right) \, d\alpha$.

Value

a single numeric value

References

Delgado M., Vila M.A., Voxman W. (1998), On a canonical representation of a fuzzy number, Fuzzy Sets and Systems 93, pp. 125-135.

See Also

Other characteristics: ambiguity, expectedValue, weightedExpectedValue, width

Other deffuzification: expectedValue, weightedExpectedValue

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, weightedExpectedValue, width

`weightedExpectedValue` *Calculate the Weighted Expected Value of a Fuzzy Number*

Description

The calculation of the so-called weighted expected value is one of possible methods to deffuzify a fuzzy number.

For $w = 0.5$ we get the ordinary `expectedValue`.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
weightedExpectedValue(object,
    w=0.5, ...)
```

Arguments

`object`	a fuzzy number
`w`	a single numeric value in [0,1]
`...`	additional arguments passed to `expectedInterval`

Details

The weighted expected value of A is defined as $EV_w(A) := (1-w)EI_L(A) + wEI_U(A)$, where EI is the `expectedInterval`.

Value

a single numeric value

See Also

Other characteristics: `ambiguity`, `expectedValue`, `value`, `width`

Other deffuzification: `expectedValue`, `value`

Other FuzzyNumber-method: `alphacut`, `alphaInterval`, `ambiguity`, `Arithmetic`, `as.character`, `as.FuzzyNumber`, `as.PiecewiseLinearFuzzyNumber`, `as.PowerFuzzyNumber`, `as.TrapezoidalFuzzyNumber`, `core`, `distance`, `evaluate`, `expectedInterval`, `expectedValue`, `FuzzyNumber`, `FuzzyNumber-class`, `integrateAlpha`, `piecewiseLinearApproximation`, `plot`, `show`, `supp`, `trapezoidalApproximation`, `value`, `width`

width *Calculate the Width of a Fuzzy Number*

Description

The width (Chanas, 2001) is a measure of nonspecificity of a fuzzy number.

Usage

```
  ## S4 method for signature 'FuzzyNumber'
width(object, ...)
```

Arguments

object	a fuzzy number
...	additional arguments passed to expectedInterval

Details

The width of A is defined as $width(A) := EI_U(A) - EI_L(A)$, where EI is the expectedInterval.

Value

a single numeric value

References

Chanas S. (2001), On the interval approximation of a fuzzy number, Fuzzy Sets and Systems 122, pp. 353-356.

See Also

Other characteristics: ambiguity, expectedValue, value, weightedExpectedValue

Other FuzzyNumber-method: alphacut, alphaInterval, ambiguity, Arithmetic, as.character, as.FuzzyNumber, as.PiecewiseLinearFuzzyNumber, as.PowerFuzzyNumber, as.TrapezoidalFuzzyNumber, core, distance, evaluate, expectedInterval, expectedValue, FuzzyNumber, FuzzyNumber-class, integrateAlpha, piecewiseLinearApproximation, plot, show, supp, trapezoidalApproximation, value, weightedExpectedValue

Index